Compressive Sensing-Based IoT Applications: A Review

Hamza Djelouat * , Abbes Amira and Faycal Bensaali

College of Engineering, Qatar University, Doha 2713, Qatar; abbes.amira@qu.edu.qa (A.A.);
f.bensaali@qu.edu.qa (F.B.)
*   Correspondence: hamza.djelouat@qu.edu.qa

check for
updates

Abstract: The Internet of Things (IoT) holds great promises to provide an edge cutting technology that enables numerous innovative services related to healthcare, manufacturing, smart cities and various human daily activities. In a typical IoT scenario, a large number of self-powered smart devices collect real-world data and communicate with each other and with the cloud through a wireless link in order to exchange information and to provide specific services. However, the high energy consumption associated with the wireless transmission limits the performance of these IoT self-powered devices in terms of computation abilities and battery lifetime. Thus, to optimize data transmission, different approaches have to be explored such as cooperative transmission, multi-hop network architectures and sophisticated compression techniques. For the latter, compressive sensing (CS) is a very attractive paradigm to be incorporated in the design of IoT platforms. CS is a novel signal acquisition and compression theory that exploits the sparsity behavior of most natural signals and IoT architectures to achieve power-efficient, real-time platforms that can grant efficient IoT applications. This paper assesses the extant literature that has aimed to incorporate CS in IoT applications. Moreover, the paper highlights emerging trends and identifies several avenues for future CS-based IoT research.

Keywords: Internet of Things (IoT); compressive sensing (CS); hardware implementation; reconstruction algorithms

1. Introduction

Internet of Things (IoT) refers to the large set of smart embedded devices connected to the Internet to provide specific services to meet the users demands [1]. IoT presents the shift from only connecting the end-user devices to the Internet to using the Internet itself to interconnect smart objects (known also as IoT devices) and to communicate with each other and/or with humans to offer a wide range of applications and services [2,3].

IoT platforms usually deploy a large scale of smart objects including wearable sensors, actuators and radio frequency identification (RFID) devices to remotely monitor different physical, environmental and physiological quantities to improve the everyday activities of the end-users [4,5]. In fact, the IoT devices often operate in a long-term mode and communicate wirelessly with each other and with a central fusion node in order to empower diverse remote monitoring platforms. In general, remote sensing devices are battery-driven, hence their performance is prone to the limited battery lifetime leading to both poor integration and user adherence. Thus, to overcome those limitations, the acquired data should be first compressed and then transmitted through optimized paths to the fusion centre to minimize the high energy. However, applying advanced data compression and transmission techniques may also consume considerable onboard energy. Therefore, the adopted compression technique has to sustain a long-term efficient monitoring along with an optimized power consumption.

Compressed sensing (CS) is an emerging signal processing paradigm which aims to acquire directly a compressed form of signals with sparse behavior at the sensing stage and enables a high reconstruction quality at the receiver stage [6,7]. CS presents an alternative paradigm to the traditional acquisition fundamentals that state that the number of measured samples should be at least equal to the number of samples in the original signal to ensure an exact recovery. However, these conditions do not take the structure of the signal into consideration. Thus, if the signal of interest is sparse, i.e., the signal can be represented by a smaller number of non-zero coefficients than its original dimension, CS claims that taking only a few numbers of random linear measurements (projections) of the sparse signal is enough to capture the salient information in the signal to provide an acceptable reconstruction quality. CS aims to shift the complexity from the sensors which are usually resources constrained and self-powered to the receiver side which is usually installed on computing platforms with relaxed constraints.

For most real-world applications, it is always possible to find a sparse or compressible representation for signals of interest using the appropriate transformation. Thus, CS has spread widely in various applications such as radar, image processing, bio-signal compression, wireless communications and many others.

For instance, the healthcare sector has witnessed a tremendous efforts to explore CS in different applications. Experts believe that CS would be beneficial for a wide range application in medical laboratory and pathology testing, particularly where many data are generated. CS may also improve wearable health monitoring platforms, making it smaller, cheaper, and more energy efficient [8]. CS is expected to optimize power and energy used in wireless ambulatory devices, hence extending the sensor lifespan and significantly simplifying hardware design and reducing both the size and cost of the entire healthcare platform. CS-based healthcare applications include medical imaging [9], electrocardiogram (ECG) monitoring [10], EEG compression [11], biometric solutions [12], etc.

Subsequently, IoT platforms have also witnessed the integration of CS into its various application based on two properties that most of the IoT platforms exhibit. First, a wide range of real-world data can be well approximated by a sparse signal using the appropriate transform. For instance, both discrete cosine transform (DCT) and discrete wavelet transform (DWT) provide a good sparse representation for ECG , images, temperature, humidity data, etc. Moreover, it is always possible to form a sparsifying basis by means of dictionary learning methods [13]. Thus, exploring CS has been widely investigated in long-term data acquisition for large-scale wireless sensor networks (WSNs) [14–16].

The second property relies on the sporadic transmission schemes that most of the IoT platforms exhibit. In a sporadic transmission scheme, not all devices transmit their data simultaneously to the fusion node, rather only a small number of devices contribute on the aggregated signal at any given time. i.e., the rate of the active devices at each transmission time slot is very small. Thus, the architecture sparsity can be explored and the knowledge about the nodes activity can be exploited at the cloud level by means of sparsity aware and joint detection protocols to achieve a high data reliability with a small number of transmitting devices.

The objective of this paper is threefold: First, it aims to present a simple and a coherent overview of the fundamentals and the mathematical models that underlie CS concept as well as its well-considered recovery algorithms. Further, the paper highlights the main difference between CS and state-of-the-art compression techniques. Additionally, the paper also reviews the distributed compressive sensing (DCS) which represents the extension of CS to the multi-channel acquisition scenario.

Secondly, the paper surveys the different research efforts in the area of CS-based IoT platforms aiming to emphasize on the importance of deploying joint CS-IoT applications. Since IoT platforms are multidisciplinary paradigms that support the connection between different cross layers from the data sensing to the enabled services and applications. The paper divides the IoT into three main layers, namely, sensing, processing an application. Thereafter, for each layer, the paper delivers a thorough comprehensive discussion for the extant efforts and research directions that have been investigated and

quantified in the literature to design and empower a high efficient embedded CS-based IoT platforms. To the best of our knowledge, no such review study has been presented in the literature before.

Third, the paper extends the discussion to illuminate the main research orientations and the emerging trends that have been originally developed out of the IoT scope. However, they hold great potentials for future integration within the context of energy-efficient real-time response CS-based IoT applications. This review paper aims to present a significant contribution to the joint CS-IoT researches by reviewing current work, diffusing insights and highlighting the important research trends that can be devoted to the CS-based IoT applications.

The remaining sections of this paper are organized as follows. Section 2 provides an overall description of general IoT framework and architecture. In Section 3, a brief overview of CS concept is provided by describing the system model, highlighting the general considerations on the design of the sensing matrix and presenting the most considered recovery algorithms associated with CS. Section 4 addresses the main issue of this paper by discussion thoroughly the joint CS-IoT efforts over the different IoT layers. The limitations facing each layer are exposed and the related CS efforts that have been made to tackle them are analyzed. Section 5 presents a discussion about open research issues and avenues for future work in IoT. Section 6 concludes the paper.

2. IoT Framework

The international telecommunication union (ITU) defines the IoT as a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies [17].

IoT consists of a network of sensors, actuators, wireless communication protocols and data processing technologies that interact with each other to provide a specific application [18]. IoT is often characterized by a large number of highly dynamic heterogeneous devices each with different and limited communication and computation resources. This heterogeneity in both the software/hardware levels requires a new level of networking/communication protocols as well as adaptation mechanisms [19]. Moreover, several other issues have to be addressed as well in both the integration and the management of these devices, such as scalability, information exchange, power consumption, interoperability and system flexibility for the dynamic change in the network topology [20]. Moreover, IoT has shifted the applications from the scale of a single device to real-time massive deployments of embedded cross-platform and cloud technologies. Therefore, to link everything together, several research groups and standardization bodies have approached and addressed these issues from several perspective [21].

Subsequently, different architectures have been proposed in the literature to establish a universal framework for IoT applications. These architectures take into consideration several parameters related to the IoT devices, the communication protocols, the networking and the targeted applications and services. For instance, a three-layer IoT platform consisting of a perception layer, a communication layer and a service layer has been proposed in [22,23]. Liu et al. considered an IoT architecture of four layers, namely physical, transport, middleware and application layers [24]. A five-layer IoT platform consists of perception layer, processing layer, transport layer, application layer and the business layer has been proposed in [25]. Table 1 summarizes a three-layer IoT platform with a brief description of the functionality of each layer. The IoT platform presented in Figure 1 consists of several sensors that can be deployed to gather information about the different physical phenomenon, these sensors are interconnected through a wireless network and communicate with a local processing unit where the data can be stored and lightweight processing can be performed. Afterwards, data are routed to the application layer hosted on the cloud where different data analytic algorithms can be applied to provide explicit services.
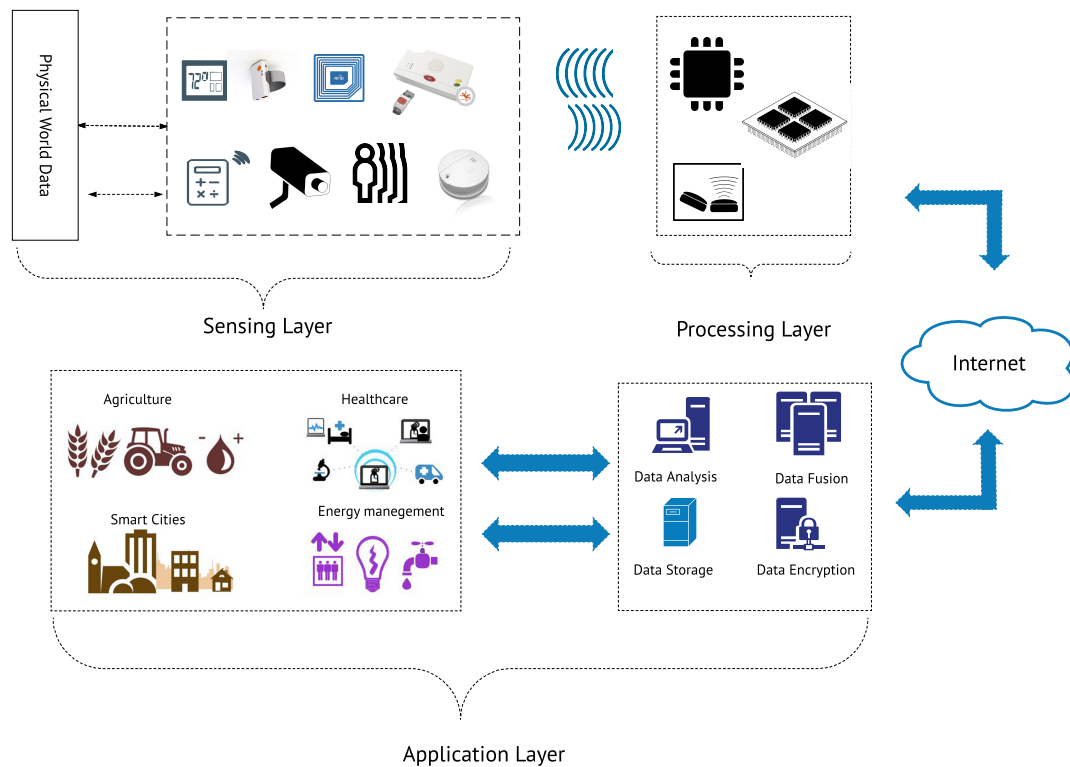
Figure 1. General IoT platform architecture considered in this paper.

Table 1. A three-layer architecture for IoT platform.

IoT Layer	Description
Sensing layer	Collect physical data entities (temperature, ECG blood pressure) using different sensors and convert the collected measurements into digital signal.
Processing layer	The Processing Layer provides mainly transient data storage for the data received from the sensors and performs local data processing, executes real-time actions, and up-links the data to the cloud.
Application layer	The task of the Application Layer is based on the information routed from the processing layer to perform data analysis and develops diverse IoT services.

IoT platforms have evolved rapidly in the last decade by leveraging the interactions among different types of smart devices, such as vehicles, medical sensors, cameras and RFID systems. Subsequently, a wide range of applications has been enabled, for instance, experimental monitoring, automated industry, connected health-care, smart buildings, intelligent transportation systems (ITS), smart grid, etc. [26]. Moreover, growing interest has been dedicated to incorporate IoT platforms in an environmental context. Strong efforts have been made by several national governments and leading information technology (IT) companies to adopt IoT solutions in the development of smart cities [27]. Even though a unified smart city framework has not been clearly defined yet, all the effort to develop this sector of applications aims to fully exploit any public resource to enhance the quality of the services offered to the citizens [28]. Smart cities can provide assistance for public service management. These services include transportation [29–31], city surveillance and emergency management [32,33], preservation of cultural heritage [34] and environment monitoring [4,28,35].

3. Compressive Sensing Overview

CS is a two-stage paradigm that acquires the data in a compressed form at the sensor and reconstructs efficiently the original data from a fewer sample signal at the receiver. Thus, CS simultaneous senses and compresses the data in one single operation.

The well-established traditional sensing and processing theory in WSNs relies mainly on the Nyquist–Shannon sampling theorem which presents the main pillar of the digital processing realm. This theorem states that, given any signal with a bandwidth of W, its information is entirely preserved if it is sampled at secured sampling frequency $f_s \geq 2W$. Further, with the ever-increasing number of digital sensing devices, several challenges have faced this paradigm, from the torrent amount of data generated by the different sensing devices to the high data transmission rate and unbearable constraints on the sampling/processing platforms. Such challenges would largely increase the cost of the wireless transmission and shorten the sensors lifespan. Thus, to address the challenges associated with high-dimensional data, researches depends often on compression techniques by discarding the redundant samples in the signal. The benefits of compression will apply to the whole ubiquitous computing environment, as decreasing data size means shorter communication delays, efficient usage of bandwidth and reduced battery drain. For instance, taking WSN applications, where the energy consumption in the sensing nodes is mainly dominated by wireless communication rather than the onboard processing. State-of-the-art radio transmitters exhibit energy consumption of order of nJ/bit while all the other sensor circuits consume at most only tens of pJ/bit [36]. This cost disparity suggests that some data reduction strategy at the sensor node should be employed to minimize the energy cost of the system. Thus, reducing the number of the transmitted samples is necessary to achieve a long lifespan. However, it should be noted that these compression techniques remain useful as long as they do not consume more energy than the one saved by reducing the transmitted samples. In fact, in WSN, this condition is fulfilled.

Recently, to tackle the issue of acquiring a huge amount of redundant samples, a very interesting theory, namely CS, has been proposed [6,7]. CS suggests that it is possible to surpass the traditional limits of sampling theory for a specific type of structured signal by reducing the sampling rate without any significant loss in the data information. The authors argued that, if the information rate of the signal is less than its bandwidth rate, i.e., the signal is either sparse or compressible, then it is possible to capture all the information without any loss using fewer samples than what the Nyquist–Shannon theorem stats. Hence, CS enables a potentially large reduction in the sampling and computation costs for sensing signals with sparse or compressible behavior.

CS differs significantly from state-of-the-art compression techniques. While the latter exhibit a two stages framework, first, the signal is sampled with respect to its bandwidth rate following the Nyquist sampling frequency high f_s to acquire an N-length signal. Secondly, data compression is performed by means of filtering, coding and extracting the most salient features in the signal to reduce the signal dimension from N to M such that $(M \ll N)$. CS, on the other hand, is a one single stage framework that aims to perform both acquisition and compression simultaneously to acquire directly a compressed form of the signal. To this end, CS leverages the fact that most real-world signals exhibit a sparse behavior when represented by the appropriate basis (this basis can be either a fixed transform such DCT and DWT or it can be formed by means of dictionary learning techniques). A sparse behavior indicates that the signal information rate is much smaller than its bandwidth rate. Thus, rather than sampling the signal at a Nyquist rate and then performing compression, CS takes random measures of the signal in a sub-Nyquist rate corresponding to the signal information rate. This process is performed by taking inner products between the signal of interest and a well-designed tall sensing matrix $\mathbf{\Phi} \in \mathbb{R}^{M \times N}$. Therefore, to reduce sampling rate and to achieve the objective of joint sensing and compressing, CS is applied prior to sampling and quantization adopting two approaches, Analog CS and Digital CS [37]. Therefore, Applying CS will result in the sensing/transmission of fewer samples. Thus, such paradigm is well suited for WSN applications which require a continuous data.

3.1. Mathematical Overview

Let the orthogonal basis $\{\Psi_i\}_{i=1}^N$ span $\mathbb{R}^N$. Then, any signal $x \in \mathbb{R}^N$ can be expressed as a linear combination of the elements of $\mathbf{\Psi}$ with the elements of the vector $s = [s_1, s_2, \cdots, s_N]^T \in \mathbb{R}^N$ such that $x = \sum_{i=1}^N \Psi_i s_i$. The signal x is said to be k-sparse if the vector s has only $k \ll N$ non-zero entries.

The set of the indices corresponding to the positions of the non-zero entries of s is called the support of s and denoted as Σ_k. In addition, the matrix $\boldsymbol{\Psi}$ is called the sparsifying matrix in the context of CS.

To perform a compressed acquisition on the data, the input signal x is multiplied by a tall random sensing matrix; hence, the acquisition process in CS can be modeled by:

$$y = \boldsymbol{\Phi}x = \boldsymbol{\Phi}\boldsymbol{\Psi}s \tag{1}$$

where $\boldsymbol{\Phi} \in \mathbb{R}^{M \times N}$ represents the sensing matrix used to acquire and compress the data. Moreover, the ratio $\frac{N}{M}$ is defined as the compression factor (CF).

3.2. Sensing Matrix

Ideally, the sensing matrix $\boldsymbol{\Phi}$ has to fulfill the following constraints:

- Optimal or near-optimal reconstruction performance: The measured data maintain the salient information of the signal for reconstruction purposes.
- Optimized sensing performance: Only a few measurements are required to obtain an optimal (near-optimal) recovery.
- Universality: The sensing matrix maintains a low coherence with almost all sparsifying matrices.
- Low complexity, fast computation and structure-based processing: These features of the sensing matrix are desired for large-scale, real-time sensing applications.
- Hardware friendly: Easy and efficient implementation on hardware is necessary.

The above-mentioned conditions present the ideal criteria for any sensing matrix. However, it is hard to find matrices that satisfy all these conditions. Thus, two relaxed conditions that can grant a high reconstruction quality have been established for designing the sensing matrix. First, a low coherence between the sensing matrix and the sparsifying matrix is recommended to enable the sensing matrix to capture the salient information of the original signal with the minimum number of projections [38]. Second, the sensing matrix should satisfy the restricted isometry property (RIP) to ensure that the compression preserves the signal information (k-RIP) and to ensure that mapping the signal x from a high dimension space $\mathbb{R}^N$ to a lower dimension one $\mathbb{R}^M$ should be unique, i.e., it should preserve the distance between each pair of two distinct signals x and x' ($2k$-RIP) .

3.3. Reconstruction Algorithms

Fast and efficient reconstruction algorithms are the keys to incorporate CS in real-world applications, thus developing such algorithms have been the main concern in the CS community. Several reconstruction algorithm classes have been proposed in the literature. Nevertheless, the well-recognized algorithms fall under the umbrella of two major algorithmic approaches, namely, convex optimization and greedy algorithms.

3.3.1. Convex Optimization

A straightforward approach to recover a sparse vector x is to solve the $\ell0$ minimization problem:

$$\hat{x} = arg \min \|x\|_0 \quad subject \quad to \quad \mathbf{y} = \boldsymbol{\Phi}x \tag{2}$$

However, the $\ell0$ minimization problem is considered as an NP-hard for large scale matrix [39], i.e., solving Equation (2) for any large matrix $\boldsymbol{\Phi}$ is necessarily computationally intractable. Therefore, convex relaxation to Equation (2) has been considered to overcome the drawback of $\ell0$, leading to $\ell1$-minimization.

The $\ell1$-minimization approach, known as basis pursuit (BP) [40], considers the following solution:

$$\hat{x} = arg \min \|x\|_1 \quad subject \quad to \quad \mathbf{y} = \boldsymbol{\Phi}x \tag{3}$$

A noisy acquisition model adapts itself by considering the basis pursuit denoising (BPDN) solution [41]:

$$\hat{x} = arg \min \|x\|_1 \quad subject \quad to \quad \|\mathbf{y} - \boldsymbol{\Phi}x\|_2 \leq \epsilon \tag{4}$$

where ϵ represents the upper bound on the noise level.

Further, if there is no knowledge about the noise level, least absolute shrinkage and selection operator (LASSO) approach [42] can be explored:

$$\min_{\hat{x}} \frac{1}{2}\|\boldsymbol{\Phi}\hat{x} - \mathbf{y}\|_2 + \gamma\|\hat{x}\|_1 \tag{5}$$

where $\gamma > 0$ represents a tuning parameter for different levels of sparsity.

3.3.2. Greedy Algorithms

Greedy algorithms have been widely exploited in CS applications due to their relatively simple framework which provides a fast reconstruction with a low implementation cost. These methods enhance iteratively the approximation for the signal by making locally optimal choices. Greedy algorithms consist of two main steps, element(s) selection and coefficients update. These methods are usually initialized with a residual $r^{[0]} = y$, signal estimate $\hat{x}^{[0]} = 0$ and empty support set T = ∅. At each iteration j, single or multiple elements from the sensing matrix are added to the support set, the signal estimate is calculated using a least square approach $x^{[j]} = \boldsymbol{\Phi}_T^{+}y$. Additionally, the residual is minimized $r^{[j]} = y - \boldsymbol{\Phi}_T x^{[j]}$. The algorithms halt when the residual norm is smaller than a predefined threshold.

Currently, the well established greedy algorithms include gradient pursuit [43], matching pursuit (MP) [44,45] and orthogonal matching pursuits (OMP) [46]. OMP offers a fast recovery compared to convex optimization approaches, yet it suffers from bad recovery quality for signals with a low degree of sparsity. Thus, several improved versions of OMP have been proposed, such as compressive sampling matching pursuit (CoSaMP) [47], subspace pursuit (SP) [48], Regularized OMP [49], Stagewise OMP [50] and orthogonal multiple matching pursuit [51].

The performance of these recovery algorithms depends on the targeted application and no unique recovery metric is established to determine the best recovery technique for all scenarios. Thus, both theoretical and experimental performance comparison between the different classes of CS recovery algorithms can be found in [52–55]. Table 2 lists the details regarding the complexity and the minimum number of measurements required for each algorithm to achieve its optimal recovery bound.

Table 2. Complexity and minimum measurements required for CS reconstruction.

Algorithm	Minimum Number of Measurements	Complexity
BP	$k \log N$	$\mathcal{O}(M^2 N^{1.5})$
OMP	$k \log N$	$\mathcal{O}(kMN)$
CoSaMP	$k \log N$	$\mathcal{O}(MN)$
SP	$k \log \left(\frac{N}{k}\right)$	$\mathcal{O}(MN \log k)$
Stagewise OMP	$N \log N$	$\mathcal{O}(N \log N)$

3.4. Distributed Compressive Sensing

Conventional CS theory explores only the intra-signal structures (sparsity and compressibility) at a single sensor. However, if the scenario of interest presents an architecture where the data are not only collected by a single sensor but rather with a distributed network of sensors, then exploiting the collaboration of these sensors would lead to an additional gain in the information collected and improve the design of the application. Such scenario arises in several IoT monitoring applications, where a network of wireless sensors is deployed to measure different scalar physical/physiological quantities, such as temperature, light, humidity, human vital signs, etc. If the sensors are densely deployed, their measurements are correlated and they are likely to share certain structures.

CS-based multi-sensor architecture with *J* sensors can exhibit two scenarios. In the first one, the acquired signals are independent; hence, the reconstruction problem is equivalent to performing *J* individual reconstruction operation for each sensor and no collaboration between the sensing nodes can be explored for the reason that each sensor holds no information about the data acquired by the other sensors. However, in the second scenario, which emerges frequently in most IoT applications, the signals acquired from all the sensor are highly correlated, which presents an inter-signal sparsity structure. Therefore, to reconstruct the whole signal ensemble, not only the individual sparsity of each sensor measurements can be exploited, but also the dependency among the samples acquired by different sensors at the same time should be explored to reduce the required number of measurements to assemble the most pertinent information related to the sensed phenomenon. This process ,which we refer to as joint measurements recovery, is the motivation for introducing distributed compressive sensing (DCS) concept [56].

DCS presents a new distributed coding paradigm that exploits both intra- and inter-signal correlation structures. In DCS scenarios, a number of sensors acquire sparse signals which are correlated with each other. Each sensor individually acquires a compressed signal by taking random projections and then transmits it to the fusion node. More importantly, DCS requires no collaboration between the sensors during the acquisition phase. However, the recovery process can exploit the inter-signal correlation between the measurements to reconstruct all the signals simultaneously. The recovery algorithms associated with DCS are derivatives from their CS counter-parts, such as multichannel-BPDN [57], simultaneous OMP (SOMP) [58] and distributed compressive sensing OMP (DCS-SOMP) [56] which is the general form of SOMP.

4. CS-Based IoT Applications

The heart of any IoT platform is the smart devices that build up the sensing layer. Data acquisition is performed using several sophisticated smart IoT devices that are deployed in different locations to collect various types of data over a long period depending on the targeted application. The collected data are usually large and contain some redundant information. Therefore, the data are first transmitted to a local processing unit with adequate storage and computing capacities to perform different pre-processing techniques on the samples to extract different information and features. The processing unit also plays the role of a gateway to route only the useful information to the cloud rather than transmitting the whole collected data, this approach can significantly reduce the network bandwidth.

CS presents a very promising paradigm to be explored in IoT applications. A CS-based IoT framework can be implemented on three levels. First, at the sensing level, where data acquisition and transmission process presents a challenging task in term of power consumption, thus minimizing the latter is the critical issue to tackle. Therefore, CS can be deployed effectively as a compression technique to develop an energy-efficient scheme for data acquisition and transmission. Second, CS can be deployed on the embedded systems implemented on the local processing unit to realize "CS-based Edge computing platform" where the compressed data transmitted from the sensors can be aggregated, stored and reconstructed to extract salient information and important features to be sent to the cloud which hosts the application layer. The latter represents the third level in which CS can be explored along with other data analytic algorithms to execute actions and provide solutions based on the specific tasks that the IoT is supposed to deliver. Figure 2 illustrates the possible scenarios where CS can be deployed over IoT platform.
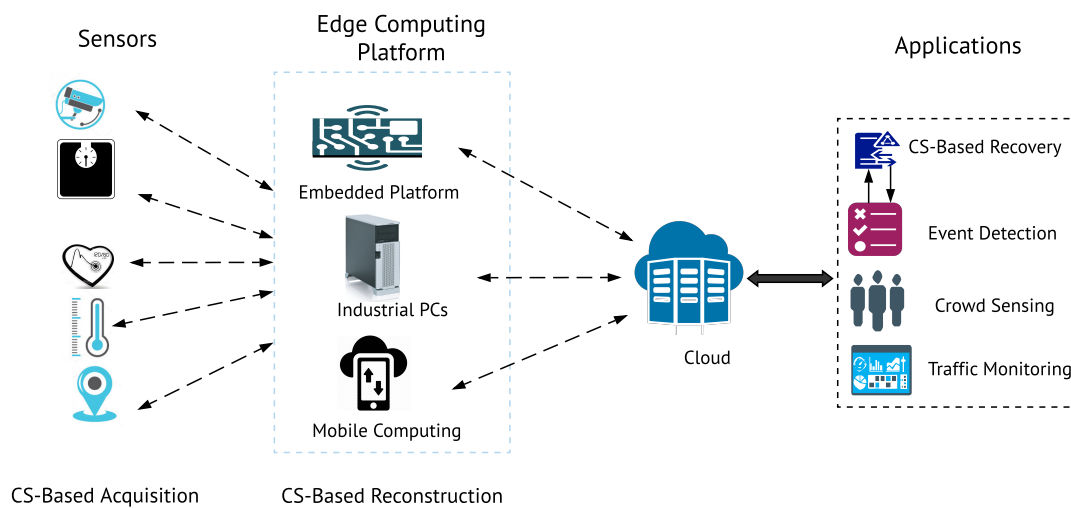
Figure 2. CS-based IOT platform.

4.1. Sensing Layer

The proper integration of sensors, actuators, RFIDs and communication technologies will deliver a strong foundation for the sensing layer of any IoT platforms. An adequate sensing layer that meets the high demands of energy-efficient IoT application should be empowered by exploring efficient hardware/software solutions. The latter consist of incorporating state-of-the-art compression techniques and data routing protocols.

The hardware part of the IoT sensing layer is enabled using two main technologies, RFIDs and WSNs. While RFIDs are mainly used to enable low-cost identification and tracking applications, WSNs can provide the IoT with broad sensing and actuation capabilities. In addition, intensive research and efforts have already been dedicated to expanding the WSNs to cover unlimited set of IoT applications.

RFIDs generally consist of two major elements, an RFID tag embedded on a chip with a unique identification sequence (ID) and a radio scanner device, called RFID reader. RFIDs have gained a lot of attention in IoT applications, they serve to store data, to track objects and to communicate with other devices [59]. Moreover, RFIDs present appealing solutions due to their ability to operate in battery-free mode and their lowest cost and small dimension [23].

For IoT applications, RFIDs have been used in different variations, the tags could be implemented on the human body [60], deployed on the skin [59] or attached to the walls or other objects. Although the reading distance of RFIDs (the distance between the tags and the reader) is not large (5–6 m), they can be deployed efficiently for indoor tracking applications [61].

Several IoT applications have witnessed the integration of RFIDs, for instance, Khan et al. [62] presented the design and the implementation of a GPS-RFID tag that can be effectively deployed for different IoT tracking applications. The designed tag operates in a semi-passive mode (battery assisted) with a reading range up to 5.6 m. In addition, it can be reconfigured to serve as a continuous data transmitting platform in the online mode or as data logging platform in the offline mode. Attran et al. [63] presented new chipless RFID tag using microelectromechanical (MMES) systems technology for IoT applications, the proposed prototype is expected to reduce the cost of the RFID tag as the chip fabrication is eliminated, hence, the cost of the RFID tag will be as low as barcode labels.

On the other hand, WSN-based IoT applications can be divided into two categories: outdoor and indoor. Outdoor monitoring collects environmental data that can be used in the context of smart cities (light, humidity, traffic, GPS tracking, and air monitoring), whereas indoor monitoring is more concerned with smart-home applications that can provide remote healthcare services, in which the

sensors can be either distributed over the house to measure humidity, temperature and detect motion or they can be deployed over the human body to acquire vital signs such as electrocardiogram (ECG), electroencephalography (EEG), blood pressure, glucose level in the blood, etc.

It should be noted that most of the progress made in the WSN-based monitoring applications can be adapted easily for IoT. However, there is a crucial difference between most IoT applications and their WSN counterparts in terms of device heterogeneity, i.e., WSNs usually deploy a set of similar sensors that collect a single type of data, whereas, in IoT applications, a large scale of sensors acquire different types of data. Moreover, inadequate deployment of WSN nodes can result in a severe degradation in the reliability of the system and increase the complexity and the cost for the overall platform. Thus, numerous researches that aim to bridge the gap between exciting WSNs and future IoT platforms have been widely discussed in the literature [64–67].

The major problem to tackle in the sensing layer is the high power consumption. Following the several results presented in the literature, the radio frequency transmission of the data over the wireless channel is the most power consuming process [36] and it can highly decrease the performance of the system, mainly if the sensors used are self-powered. Thus, determining the best compression technique in term of minimizing power consumption is the first step to develop an energy-efficient platform.

Besides CS, many research groups have developed low power, low-cost compression techniques exploring state-of-the-art compression algorithms [68–70]. Subsequently, various efforts have been made to establish different frameworks to compare CS with state-of-the-art compression algorithms for different applications to determine the optimum compression technique.

First, Mamaghanian et al. [71] developed a non-adaptive CS-based platform for ECG compression on the Shimmer mote. In addition, they provided a comparison study between the performance of CS and DWT-based ECG compression in terms of reconstruction quality, node lifetime and CPU execution time. Although the results show that DWT outperforms the non-adaptive CS in terms of reconstruction quality (to achieve very good reconstruction quality, DWT requires $M = 0.3 \times N$ samples in the compressed signal, whereas, to achieve the same reconstruction quality, CS requires $M = 0.5 \times N$), CS-based compression has shown to provide a better energy-efficient performance, with a node lifetime extension of 37.1% over DWT and an execution time 20 times faster than the one taken by DWT. Chen et al. [36] implemented and quantified two different implementation approaches for CS, namely, digital CS and analog CS. The analysis showed that for real WSN applications, digital CS provides a better energy efficiency performance. Moreover, they compared CS with the Lempel–Ziv–Welch (LZW) compression technique; the CS compression system offers six times higher compression factor and over ten times lower implementation (storage/power) cost.

Moreover, Razzaque et al. [69] provided a more detailed comparative study including CS, DCS, transform coding (TC), predictive coding (PC) and adaptive sampling (AS) approach [72]. The work has investigated three different datasets including temperature, seismic signals and CO_2 emission data. The overall obtained results reveal that CS improves the energy saving by a factor of 79.4% compared to 62.43% and 34% obtained by TC and AS, respectively. In addition, Abate et al. [73] conducted a comparative study in terms of reconstruction quality and computation time between CS and the segmentation and labelling technique [74] which has been considered for the definition of the new standard of the IEEE 1451 [75]. The obtained results show that, even though CS is more complex, it outperforms remarkably the segmentation and labelling technique in terms of data reconstruction and robustness to noise. Furthermore, a comparison between CS and transform coding (TC) and transmission without compression in terms of energy consumption is presented in [15]. First, the authors established energy dissipation models for both approaches to investigate the energy consumption in a unified framework. The obtained results indicate that CS extends the network lifetime up to four times compared to no transmission without compression processing if the data are highly sparse and the transmission range is between 50 m and 100 m. All these results prove that incorporating CS in IoT applications will provide more improvements in terms of energy-efficiency. Thus, with the relative superiority of conventional CS over classical compression techniques being

established, a fundamental question is how to raise the standards and further enhance the exciting results of CS to minimize power consumption. To this end, various approaches have been proposed in the literature to answer this fundamental question.

4.1.1. Adaptive Measurements

The intuitive solution is to minimize the number of the transmitted samples from the sensors to the processing unit by selecting a high CF. However, using a high CF can degrade the performance of data reconstruction. In addition, the optimal value of CF depends heavily on the signal sparsity which may vary in time (at the level of each sensor) and in space (sensors may have different readings for the same signal), hence CF can be high as long as it assures an acceptable recovery performance to meet the desired quality of service (QoS) for a specific application. Therefore, determining the optimal CF has been the quest of different studies by exploring different ideas and approaches. Fragkiadakis et al. [76] proposed an adaptive framework for selecting CF value to overcome the signal sparsity level variations from data-block to another. The proposed framework consists of a network of sensors, where each sensor transmits its data to a central node (CN). First, each sensor transmits a part of its data which can be seen as a training set to the CN. The latter starts a compression/decompression operations with several values of CF to determine the signal sparsity along with its best CF based on a predefined QoS metric (error of reconstruction). In addition, the CN creates a profile that assigns to each range of sparsity the best CF that renders the minimum reconstruction error. Furthermore, Charalampidis et al. [77] proposed a new approach to detect the changes in the signal sparsity using change point methods (CPM) [78] to update the CF value each time the signal sparsity level changes.

4.1.2. Weighted Measurements

Another approach to optimizing the number of transmitted samples is to assess a weight to each sensor depending to its importance in the application (a scenario that often occurs in the heterogeneous WSNs). This idea was first introduced in the context of CS for image reconstruction using wavelet decomposition where different CFs are assigned to different wavelet sub-bands [79]. Subsequently, borrowing the same concept to IoT applications, Lee et al. [80] introduced weighted CoSaMP (wCoSaMP) reconstruction algorithm to enable a new framework for data acquisition in heterogeneous IoT platforms where different type of sensors have different impact on the overall system, for instance, for IoT-based connected health, the ECG data are more crucial than humidity and temperature data. Applying this concept, fewer measurements are needed to achieve the same performance as conventional CS.

4.1.3. CS-Based Data Gathering

Efficient data helps to find suitable solutions for the high energy consumption in large-scale WSNs. Luo et al. [81] proposed a compressive data gathering (CDG) approach to extend the sensors lifetime in multi-hope architecture, where each node performs a CS acquisition and transform its random measurements to the next node. Subsequently, the data at the sink node will be the sum of the different random measurements. CDG approach disperses both communication and computation costs to all sensor nodes, resulting in a natural load balancing. Moreover, based on CDG, Xu et al. [82] presented compressed sparse function (CSF) algorithm for data gathering in WSNs. This approach adopts DCT to sparsify the collected data from each sensor and then transmits the sparse functions to the sink. Then, the reconstruction is performed by means of polynomial interpolation techniques. This approach allows reconstructing the whole ensemble data from an incomplete set of received measurements. Beside CSF and CDG, Xiang et al. [83] proposed minimum energy compressed data aggregation (MECDA) approach. The algorithm determines the best routing path from the sensor nodes to the sink node that consumes the minimum power. To determine this optimum path, the algorithm exploits minimum spanning tree and shortest path forest techniques. Nevertheless, this algorithm maintains its optimum performance as long as the network topology remains fixed. In addition, an opportunistic

routing approach called compressive data collection (CDC) has been proposed in [84]. In the proposed approach, each sensor selects randomly one of its neighbours by means of opportunistic routing protocols [85] and forward to its compressed reading.

4.1.4. Sparse Networks

From another perspective, IoT applications are usually deployed using dense WSNs in which the data collected from the sensors are redundant and highly correlated. Thus, exploring the redundancy in the reading each of sensor can be used to minimize its duty cycle, thus, prolonging its lifespan and subsequently the network lifetime. Furthermore, the high correlation between the measurements can be used to enable a scheme where only a few sensors operate in an active mode for each time slot. By taking advantage of these facts, Du et al. [86] presented CS based activation scheme for water distribution in IoT networks, the proposed method selects a few numbers of sensors to send their data to the sink at each time slot, which renders a remarkable reduction in the power consumption. Besides, a recent active node selection framework aims to improve signal acquisition performance, network lifetime, and the use of spectrum resources has been proposed in [87]. The proposed approach exploits the temporal correlation to select the active nodes based on the support of the data reconstructed in the previous time slot. In addition, the correlation between the sensor's measurements were exploited in [88], in which a 1-bit DCS-WSN framework was proposed to achieve more data compression while retaining an acceptable performance using the joint sparsity between the sensors reading. Zhang et al. [89] presented another idea for saving energy in WSN based on regression and CS. The main idea is to divide the sensors into different clusters, where at each cluster only one sensor (reference) node works in a periodically sampling mode, whereas all other nodes in the cluster exhibit a CS-based acquisition. After receiving data from all nodes, the sink node runs prediction algorithm to roughly estimate the signal series of all nodes based on the signal of the reference node.

A comparative study has been conducted [86] to compare the performance of different data gathering approaches in term of the achieved network lifetime. The investigated scenario includes a network of 100 sensors deployed randomly over a specific area, in which only 20 sensors are active at each time slot and all the sensors transmit their data to a single node. The lifetime achieved by the approach proposed in [86] is about 5 and 2.5 times the CDG and CSF lifetime, respectively, whereas the lifetime achieved by the CDC algorithm is about 80% of the achieved lifetime using the proposed algorithm in [86].

4.1.5. CS-Based Routing Protocols

Routing and CS have been jointly addressed in [90] in which the routing path is iteratively built through a greedy choice to minimize the intermediate coherence of the measurement matrix with the sparse matrix. Moreover, based on the results obtained in [81,90], where the authors proved the benefit of their approaches in terms of good reconstruction quality with fewer transmitted samples, Caione et al. [91] introduced and quantified a new data routing approach. The authors proposed a DCS-based data gathering in large-scale WSNs based on CS acquisition, where they have investigated the performance from a node lifetime point of view. The approach considered in [91] provides each node with the ability to take the proper decision about the optimum value of its CF. After compression, the data are routed by exploring a hybrid approach based on DCS and pack and forward (PF) technique. The aim of this hybrid is to reduce the overall number of transmitted packets. The obtained results reveal an extension in the network lifetime compared to the use of conventional DCS even with a large number of sensors in the network. Moreover, Masoum et al. [92] built up a probabilistic model for the ensemble of the signal using a Bayesian approach and adopted belief propagation (BP) algorithm to recover the ensemble measurements. Their results show an improved reconstruction quality as well as a reduction in the energy consumption up to 20% compared to Multichannel BP associated with DCS.

4.2. Processing Layer

With the deployment of huge, dense WSNs to collect different types of information. Big data is generated from the sensing objects and routed to the cloud. Even though the big data provides a comprehensive knowledge about the sensed phenomena, they require very powerful storage, processing and information retrieving mechanisms. Moreover, big data is usually so massive that it exceeds the capabilities of commonly used computing environments. Thus, it restrains the performance of the IoT platforms by increasing the high-level service requirements (massive storage, expensive processing power, high latency, etc). Subsequently, a major challenge in IoT platforms design is to decide the layer in which the data can be processed. In fact, this decision depends on several factors such as real-time constraints, energy efficiency, communication bandwidth, delay, etc.

Therefore, for large-scale IoT platforms in which real-time response and power efficiency are strict requirements, opting for a device-centric processing does not seem to be effective. In a device-centric approach, the micro-controllers embedded within the sensors are exploited to process the data. Nevertheless, the computation/communication resources of these sensors are limited which would lead to poor system performances. On the other hand, opting for a cloud-centric processing can satisfy the computing requirements, however, the transmission of a substantial amount of data over the wireless channel incurs a heavy communication overhead. Consequently, cloud processing usually faces issues of scalability, high energy cost, latency and bandwidth availability [93]. Thus, a trade-off solution can be endorsed by adopting a gateway/fog centric architecture. The IoT gateway devices which are used to provide an interoperability between the different heterogeneous cross platforms are usually installed on powerful computing embedded platforms such as ARM Cortex-A [94]. In addition, smartphones can be used as processing units for gateway centric IoT [95].

Moreover, gateway/fog computing does not only bring the computing platform to the edge of the network but it also provides a low latency, location awareness, smart geographical distribution and on-line analysis [96]. Therefore, moving the computation from the cloud to the edge near the IoT devices would help to meet the computation requirements, to provide real-time response and to improve the scalability of the network. This approach is known as edge computing.

Edge computing presents the new trend in IoT infrastructure development; it aims to analyze the time-sensitive data at a platform in a proximity to where the data were initially collected rather than sending many data to the cloud. Edge computing platform does not eliminate the role of cloud computing, however, it will only send selected information and historical analysis for long-term storage and deep data analytic operations. Edge computing can be well suited for CS-based IoT applications, where the compressed data can be reconstructed at the gateway unit, then only the extracted features, the reconstruction reports and other useful service-specific information can be sent to the cloud for further deep learning, classification and objects/events detection.

Edge computing leverages the powerful local processing units to shift the data computation and storage from the cloud clusters to the embedded cross platforms. Edge computing can be accomplished using PCs, mobile computing devices, field programmable gate array (FPGAs), etc. Moreover, Edge computing offers real-time analysis and response, low-cost management as well as fewer data transferred to the cloud.

From the several available computing platforms, FPGA technology establishes itself as an appealing environment to implement such edge computing platforms. FPGAs would provide energy efficiency, interactivity, powerful computation abilities and reconfigurable embedded system to IoT. Moreover, compared to other computing platforms such as Raspberry Pi, FPGAs provide a better performance [97]. Although the implementation of CS-based edge computing on the FPGA for IoT applications have never been addressed in the literature, we present herein some of the efforts which can be explored to provide powerful computing platforms.

For any CS-based edge computing platform, efficient implementation of data reconstruction algorithms is the main task to realize. While a software implementation of these algorithms can be time-consuming due to the massive matrix multiplications requirements, an FPGA-based solution

presents an attractive alternative to accelerate the reconstruction process by leveraging the high level of parallelism of FPGA platforms.

Therefore, the hardware implementation for the different reconstruction algorithms has been investigated by the CS research community. In fact, greedy algorithms gained most of the attention compared to convex optimization approaches due to the high complexity of the latter. Nevertheless, from all the greedy algorithms, OMP has been the focus of most researchers due to its computing efficiency, stability and relative simplicity. Besides, in contrast to advanced greedy algorithms such as SP and CoSaMP, OMP does not require any inputs except for the compressed signal and the sensing matrix. OMP recovery algorithm includes two computationally expensive steps, namely, coefficients selection and signal estimation. Coefficients selection step requires intensive matrix-vector multiplication operations in order to find the closely correlated atoms to form the signal proxy. For high dimensional signals, matrix-vector multiplication is a very time-consuming process. The signal estimation step consists of solving a least squares problem which is both time and resources consuming as it includes both matrix inversion and matrix-vector multiplication. Therefore, all efforts have been made to provide an FPGA-based implementation for OMP rely on the optimization of these two fundamental steps.

Septimus et al. presented the first implementation of OMP on Xilinx Virtex-5 FPGA [98] where the implementation was validated on a signal of length $N = 128$ and a maximum sparsity of 5. Although the size of the signal and its sparsity are small, the obtained results showed that adopting FPGA-based OMP implementation has great promises for real-world applications. Furthermore, the same authors further optimized their implementation by adopting Q-R decomposition (QRD) to solve the LS problem in [99]. The obtained results revealed three times faster execution time than their first implementation. Thereafter, various efforts have been made to provide more efficient FPGA implementation for OMP. Bai et al. [100] proposed a solution to the LS problem based on QR decomposition using a vector multiplication unit (VMU) which consists of a large number of parallel computing channels. The proposed solution has been implemented on a Xilinx Virtex-6 FPGA. The authors evaluated their implementation on an image block of size 32×32 pixels achieving reconstructed quality with signal-to-noise ratio (SNR) of 23.5 dB with an execution time of only 0.63 ms. In [101] single-precision floating-point CS reconstruction engine implemented on a Kintex-7 FPGA is presented. The author fully explored the maximum hardware resources available to achieve high performance by deploying highly parallel architecture that shares computing resources among different tasks of OMP by using configurable processing elements (PEs). Rabah et al. [102] presented a descriptive analysis to OMP algorithm where they conducted a detailed study on the complexity of each step in the OMP algorithm. In addition, they adopted a Cholesky factorization method to solve the LS problem. A four-block architecture has been implemented on MATLAB-SIMULINK and Xilinx system generator (XSG) has been used to map the system-level design to a high-level description language (HDL). Subsequently, more efforts have been made to implement OMP for real-world signals. Kulkarni et al. quantified the performance of OMP implementation on FPGA Vertex-7 for ECG signal [103] and image data [104]. Recently, Quan et al. presented an efficient solution for the FPGA implementation of OMP [105]. The proposed approach uses fast Fourier transform (FFT) to implement the coefficient selection step which is represented by a correlation operation. Additional efforts for FPGA-based implementation of OMP can be found in [106–109].

Beside FPGAs, the hardware implementation of CS reconstruction algorithms on other computing platforms such as graphics processing units (GPUs) and application specific integrated circuits (ASICs) have been presented in the literature. A hardware implementation of three CS recovery algorithms on ASIC have been proposed in [110] for channel estimation in wireless network. The authors provided a comparative study of the architecture, the complexity and the cost of MP, OMP and gradient pursuit. Fang et al. [111] presented a GPU implementation of OMP where they proposed a matrix inversion update method to minimize the computation complexity of the matrix inversion problem which leads to speed up the reconstruction.

In [112], the authors presented a comparative study between the implementation of the OMP for image reconstruction using a high parallel computation with an LU decomposition for solving the LS problem. The proposed architecture has been implemented on different platforms including general purpose CPUs, GPUs, a Virtex-7 FPGA and a domain-specific many-core. The implementation results showed that reconstruction time on FPGA is improved by two times compared to the results in [99]. In addition, the GPU based implementation has shown to be three times faster than results obtained in [111].

Table 3 expatriates on the comparative results between several works presented in the literature on the implementation of OMP on different platforms.

Table 3. Comparison between FPGA implementation for OMP presented in the literature.

References	Platform	Data Size	Reconstruction Time	Max Frequency (MHz)
[100]	FPGA (Virtex6)	512	360 μs	100
[99]	FPGA (Virtex5)	128 × 128	13.7 μs	165
[105]	FPGA (Virtex7)	128	391.8 μs	165
[101]	FPGA (Kintex-7)	1024	39.9 μs	53.7
[104]	FPGA (Virtex5)	256 × 256	9.32 μs	N/A
[102]	FPGA (Virtex6)	1024	340 μs	119
[109]	ASIC	256	10.17 μs	196
[113]	GPU (NVIDIA GTX480)	81922	15 ms	N/A

4.3. Application Layer

Although CS was developed as a sensing/compression paradigm, it can be extended to other domains such as signal detection and sparse problem optimization. Thus, for IoT platform scenario, the application layer relies on the data routed from the local processing layer (edge computing platform) to provide both object-based and event-based data analytics, such as data aggregation, classification, active detection, signal identification, smart cities application, cloud status monitoring and even CS-based recovery.

4.3.1. Multi-User Detection and Identification

Regarding users/devices activity, CS-based event detection approaches exploit the fact that the rate of the active sensing devices over the overall number of the sensing nodes is usually small, which can be seen as a sparse network architecture. Therefore, CS techniques have been exploited not only to detect the signal of interest but also to detect and identify the number of the active devices/users that contribute on the data aggregated at the cloud level. Zhu et al. [114] introduced a CS multi-user detection (MUD) detectors for communication systems with code division multiple access (CDMA) by adopting BP algorithm. Furthermore, CS-based MUD using greedy CS algorithms has been proposed in [115]. The authors studied the efficiency and reliability of applying CS to detect both the users activity and to identify data in a CDMA transmission. The obtained results in [115] indicate that the orthogonal least squares (OLS) algorithm is reliable for sparse MUD even in overloaded CDMA systems. However, it is shown in [115] that the symbol errors of the OLS are mainly caused by incorrect activity detection. These results imply that improving user activity detection would enhance the overall system performance. Thus, Schepker et al. [116] introduced block-wise orthogonal least squares (BOLS) detection as sparse MUD for sporadic communication using CDMA.

In [117,118] a CS-based two-stages framework for joint data identification and detection is proposed for IoT applications using multi-carrier CDMA (MC-CDMA) with data transmitting scheme based on sparse index multiple access (SIMA) [119]. First, CS is exploited to estimate the channel state information (CSI) to determine the number of the devices/users that are actually

transmitting data using OMP. Next, using the estimated CSI, the transmitted data are detected using expectation-maximization (EM) algorithm. In addition, a proposed solution to detect the active users in large scale IoT by combining linear detection and CS recovery algorithm is presented in [120]. This approach aims to improve the receiver performance relying on linear detection and reducing the computational complexity by exploring efficient CS recovery algorithms.

Furthermore, Wang et al. [121] addressed the problem of MUD in massive IoT networks. The authors adopted structure matching pursuit (SMP) algorithm to jointly detect the common active users at all times, and then to detect the dynamic active users at each time slot individually. However, if the number of devices scales up remarkably, this method will suffer from a high system complexity and a slow reconstruction time. Therefore, Liu et al. [122] proposed a scalable CS-based MUD scheme to tackle this issue based on single measurement vector CS (SMV-CS) [123,124] which uses group OMP (GOMP) [125] to divide the data into different sub-blocks and reconstruct each sub-block individually.

4.3.2. CS-Based Cloud Storage

On the other hand, receiving big data has shifted the storing paradigm from single server to multiple mega-server and it is continuously growing. The massive number of data that are collected on a daily basis requires large-scale data processing engines to analyze. However, applications are rarely interested in raw data records. Instead, only the salient features are required for deep data analytics. CS can be explored efficiently to store the cloud data by reducing the data size and discarding the non-relevant features to facilitate different types of applications. Many IoT data analytic applications are executed by sending queries to the cloud server to retrieve data. For instance, Top-k, range, and keyword queries present the most popular types of queries for IoT and WSN applications. Top-k query means that the k highest (or lowest) data values are retrieved from a dataset [126]. Thus, by leveraging the fact that the CS compresses and preserves an approximate form of the data, Zhang et al. [127] proposed a new impression store cloud architecture based on CS that does not store all the raw data, but rather some features that can yield to provide a successful application in term of QoS. This approach saves storage capacity and bandwidth and allows for efficient and scalable parallel updates and queries. This approach consists of compressing the data to the minimum factor that allows the recovery of only the application-defined principal components. In addition, Chen et al. [128] a two-stage compression scheme. First, the data are compressed at the sensor level with an M samples. Afterwards, they wait for a query from the cloud to send only the $k \ll M$ samples that are required for a specific defined application.

4.3.3. Mobile Crowd Sensing

CS can be deployed efficiently for applications related to smart cities in order to monitor environmental and urban conditions using the so-called mobile crowd sensing (MCS) approaches. MCS explores the user's mobility; the sensors embedded within the users smart-phones and the existing wireless infrastructures are used to sense and collect environmental data. The collected data can relate to a number of phenomena, including air quality, noise level, street surface and pavement conditions. MCS consists of dividing a specific area to N different cells, where, at each cell, single or multiple users are allocated with different sensing tasks. The main drawback of conventional MCS approaches is that they incur a high sensing cost in term of energy consumption to achieve a high sensing quality. In addition, many participants are required to collect and transmit data which puts high constraints in term of bandwidth occupancy. Thus, sparse MCS presents an alternative solution to the conventional methods. Sparse MCS consists of allocating only a small number of cells ($M \ll N$) with participants to collect data. The aim of this approach is to estimate the data of the all N cells from the collected data from M cells. This can be seen as a CS basic problem by estimating an N-length data from an incomplete set of measurements. Sparse MCS approaches face two key challenges, first, how to select the cells that provide the optimum coverage, second, how to assess the estimated data quality without apriori knowledge of the data of the non-sensed cells.

To address these challenges, wang et al. [129] presented a framework for sparse MCS application with an iterative task allocation process. In this framework, cells allocation operation is determined using uncertainty-based approach [130]. After transmitting data from the selected cells, the whole area conditions are estimated using spatiotemporal CS recovery algorithms [131]. The obtained results show a significant reduction in the sensing cost while still guaranteeing the overall data quality for urban sensing. Thereafter, several efforts have been made to further enhance the performance of sparse MCS in terms of privacy [132], cell allocations [133,134] and reducing the number of participants required for data collection [135].

4.3.4. Traffic Monitoring

Traffic monitoring is the crucial task that can provide great insights for the design and the management of infrastructures in highly urban cities. The state-of-the-art traffic monitoring approaches exploit roving vehicles and the mobility of smart-phones to provide periodic reports regarding traffic status, driving speeds and the flow direction to estimate the whole traffic conditions [136]. These approaches often employ an intensive number of users (probes) to cover the roads of interest. Nevertheless, their performances are usually limited by the number of users, energy expenditure of the sensing devices, privacy as well as the high cost associated with hiring many participants to provide the complete traffic map. However, the high dimensional datasets for traffic conditions can be well approximated with low-rank matrices by mining the hidden structure of this datasets. Zhu et al. [137] used principal component analysis to mine a large set of traffic data collected in China. Their analysis showed that the energy is concentrated in just a few principal components which underpin the applicability of CS-based traffic sensing. On the other hand, individual traffic reports are usually error-prone; for instance, if the car goes through obstructions (urban canyons and underpasses), the reliability of their GPS data would fade. In addition, with the continuous mobility of the probes, multi-path propagation phenomena will degrade the quality of wireless communication which will lead to huge amount of missing data. Thus, CS can be used to estimate the entire data (including the missing one) from the partially received traffic data points. The approach proposed in [137] has proven to provide an estimation error less than 20% even though 80% of the data were not received. Furthermore, by leveraging the fact that the traffic conditions among interconnected roads are highly dependent, the correlation model between these traffic conditions can be explored to reduce the number of users, while maintaining the same quality for the entire city traffic map. Liu et al. [138] presented a multiple linear regression (MLR) method based on CS to estimate the entire road traffic from a small set of measurements. The idea is to divide the map to N blocks relying on the intersecting points. Then, only a small number of blocks $M \ll N$ are assigned with j vehicles (probes) to measure the average velocity v_j and the distance to travel each road $L_j = [l_j^1, l_j^2, \cdots, l_j^M]$. Hence, to estimate the entire road velocities from the set of observed velocities $\{v_j\}_{j=0}^{j=M}$, CS recovery algorithms can be adopted. The authors compared their approach with state-of-the-art solutions based on singular value decomposition (SVD), the results showed that the proposed method achieved an estimation accuracy up to 80% while the SVD approach achieved only a 60% estimation accuracy.

Finally, more CS-based data analytic approaches can be addressed in that future. For instance, by leveraging the reconstruction reports, data mining and incremental learning algorithms [139] can be deployed at the cloud to determine the optimal number of measurements every node has to acquire to provide the best reconstruction performance.

Table 4 provides a summary for the extant efforts to integrate CS into IoT applications.

Table 4. Summary of the Joint CS-Based approaches for IoT applications.

IoT Layer	Approach	Features	Main Attribute
Sensing Layer	Adaptive measurements [76,77]	Optimize CF value	• Sparsity change detection based on QoS [76] • Sparsity change detection using CPM [77]
	CS-based sparse Networking [86–89]	Energy efficiency High reconstruction quality	• Alternate the selection of the active sensors at each time slot [86] • Nodes activation based on data reconstruct in the previous time slot [87] • Divide network into clusters, each with only one single reference node [89].
	Weighted measurements [79,80]	Optimize CF value	• Applicable on heterogeneous IoT platform • Assign for each sensor a CF value depending on its importance in the application
	CS-based data gathering [81,83,90,92]	Data gathering Low communication cost Low error rate	• CDG for multi-hope: Compress and route to the next node [81] • CSF: each node sparsifies the data and route the sparse functions to the sink [82] • (MECDA) approach: select the best route that consumes the minimum power [83]. • CDC: each sensor selects its route based on opportunistic routing protocols [84].
	CS-based Routing [90–92,140]	Extend nodes lifetime Better signal quality	• Build routing path by choosing sensing matrix that minimize the coherence with sparsyfing basis [90] • Utilize BP algorithm to recover the ensemble measurements [92] • Build the Sensing matrix by minimizing the coherence • Hybrid routing approach using PF and DCS [91] • Random walk (RN)with DCS [140]
Processing Layer	FPGA-based OMP implementation [99,100,102–109]	Fast execution High performance	• Adopt Q-R decomposition (QRD) to solve the LS problem [99] • Implement a parallel vector multiplication unit (VMU) to solve LS problem [100] • Cholesky factorization method to solve the LS problem to [102] • FFT approach to perform the correlation operation (coefficients selection) [105]
	CPU/GPU-based OMP implementation [99,110–112]	Easy implementation	• Matrix inversion update method [111] • LU decomposition for solving the LS problem [112]
Application layer	Multi-user detection [114–118,120,121]	Active users identification Enhance data detection	• User detection in networks with CDMA based transmission [114–116] • Joint identification and detection in massive IoT with MC-CDMA [117,118,120,121]
	Cloud data Storing [126–128]	Efficient storage	• CS-based Top-k query data retrieval approach [127] • A 2-stage compression architecture [128]
	Traffic Monitoring [136–138]	Reduce number of required probes low cost	• Multiple linear regression (MLR) mode based on CS to estimate the entire road traffic from a small set of measurements • Explore the correlation between the road status to reduce required probe of entire road map estimation
	Sparse MCS [89,132–135]	Low cost Reduce energy expenditure User privacy	• Allocate small of cells with specific monitoring tasks • Estimate the whole cell ensemble information from the small set of collected data • User privacy enforcement [132] • Reduce number of required participants for data collection [135] • Minimize the number of cells to collect information [133,134]

5. Challenges and Research Trends

Exploring the techniques and the approaches presented in Section 4 would provide an efficient CS-based IoT platform that can be used to enable numerous applications. However, many studies related to CS can be further incorporated. This section provides some of the research orientations and challenges that the CS research community is trying to investigate and to address. Although these efforts are introduced out of the IoT scope, they can fit well in the design and the development of an energy-efficient, real-time and secure IoT platforms. This section highlights the advantages of incorporating such techniques in IoT applications, where four of most prominent research orientations that can be well explored to enhance the IoT performance over the three different layer are discussed. For the sensing layer, using both energy-efficient CS encoders and structured sensing matrices would decrease the system complexity, hence provide energy efficiency. For the processing layer, using gateway embedded on many-core platforms instead of single-core can boost up the speed and the quality which will help meet the real-time requirements. Finally, incorporating CS-based encryption methods helps to achieve a high level of security and privacy for specific IoT application.

5.1. CS Encoder Design

The different techniques that explore CS at the sensing layer can contribute significantly to the design of energy-efficient CS-based IoT system by using the CS as a compression technique. However, to fully exploit CS theory, the sensing nodes have to collect directly a compressed form of the signal. Therefore, it is of high importance to design CS-based sensing nodes.

In fact, one of the main attributes of CS is to shift the high complexity burden from the IoT sensing devices to the gateways that are usually embedded on platforms with much-relaxed computation, communication and energy efficiency abilities. CS is expected to reduce the power consumption of the sensing and the transmission processes. Unlike conventional compression techniques which require some processing to extract the salient information, CS encoding can be implemented in two different modes, analog or digital domains. In analog CS, the linear projection is applied in the analog domain prior to digitization, whereas the digital encoder first quantizes the signal samples and then performs the equivalent modulations using digital logic. CS-based IoT applications require sensing nodes with low-power consumption design, thus they could benefit from research that aims to develop a low rate-energy efficient CS acquisition sensors [36,37,141,142]:

- Analog CS [143,144] represents the hardware implementation of the CS acquisition model (Equation (1)) in which the signals are acquired at the sub-Nyquist rate. The CS-based acquired signal is the inner product between the input signal and M random vectors. In fact, the analog CS encoder is designed using random modulator (RM) [145]. The RM is implemented using a mixer, an integrator, and an analog to digital converter (ADC). The mixer performs the inner product between the signal and the measurement matrix in a sub-Nyquist rate. The integrator accumulates the output voltage of the mixer, and it has to be reset after each sample is taken. Finally, the signal is sampled at rate $1/M$ using the ADC. For more illustrations and implementations, the reader can refer to [146,147].
- Digital CS is performed by sampling the input signal following the Shannon–Nyquist theorem, and then performing M random modulation. In addition, a non-uniform sampler (NUS) [148] technique can be used, where the CS encoder picks an M samples randomly from the whole N dimension vector after the digital conversion. The NUS can be seen as an RM modulator with binary sensing matrix with elements $\{0, 1\}$.
- Bellasi et al. [37] examined the implementation of both analog and digital encoders and showed that an inexpensive and energy-efficient digital logic is most suitable to implement CS-based data reduction. Moreover, they investigated two scenarios that can occur often in WSN. In the first scenario, the energy consumption of storage/transmission is dominating the total power balance, thus the superior compression performance of analog CS leads to a significant advantage

over digital CS. Instead, in scenarios where signal acquisition and processing are dominant, digital encoders are indeed more energy-efficient. The results hold great promises to extend CS application of CS encoder from small WSNs to massive IoT platforms.

5.2. Structured Sensing Matrix

A crucial task in CS is to select the appropriate sensing matrix, which has a crucial impact on the reconstruction performance. Moreover, selecting inappropriate sensing matrix would lead to additional system complexity and may provide unacceptable reconstruction quality.

In CS, the sensing matrix is usually designed using a dense random matrix with entries drawn from an independent identical sub-Gaussian distribution [149,150]. Gaussian processes are widely used as sensing matrices in CS; they offer a low coherence with most of the natural basis used in signal processing [148]. Moreover, they satisfy the RIP condition with high probability [151]. However, they suffer from several limitations in term of implementation on hardware due to their dense nature. Consequently, they can be implemented using a non-chip random seed to generate all the entries or they can be generated offline and stored in a large memory [152]. In addition, they slow down the measurement process of CS because it has to deal with all non zero entries. Thus, the random matrix measurement operator must be replaced by more structured sensing architectures that correspond to the characteristics of feasible acquisition hardware.

Therefore, using structured sensing matrices would improve both the sensing and the reconstruction processes. At the sensing level, deploying sparse Bernoulli matrices with entries $\{0, 1\}$ can reduce the number of multiplication operations by replacing the inner product by just an accumulation process [153], which can be seen as an NUS approach.

Furthermore, structured sensing matrices would accelerate the reconstruction time and minimize the onboard computation on the receiver side. Therefore, incorporating such techniques aligns perfectly with the vision of providing efficient CS-based IoT platform.

Using Bernoulli matrices with entries $\{1, -1\}$ [71] can be seen as good fit for FPGA reconstruction, where the multiplication process can be replaced by data accumulation only. Several works have also focused on structured sensing matrices such as Toeplitz, circular, block diagonal and permuted block diagonal (BPBD) [154] and deterministic binary bloc-diagonal (DBBD) [152]. These structured matrices have been shown to be hardware friendly by discarding the need to implement on-chip random seed for random matrices [152].

5.3. Implementation on Multi-Core Platforms

With the fast increase in the number of IoT devices, the big data transmitted to the computing platforms will put a lot of pressure on the real-time requirements and the processing will exceed the capabilities of single-core platforms. This scenario occurs in several IoT applications that require real-time video streaming. Subsequently, single-core platforms will soon fade and be replaced with multi-core platforms that have more relaxed computation and communication features [155]. Moreover, most IoT applications would utilize the end-user personal computer and mobile as the edge computing platform; thus, the implementation of reconstruction algorithms on multi-core platforms that are similar to the ones held by the end-user has been addressed in the literature for different applications. For instance, taking advantage of parallel computing, Borghi et al. presented in [106] the design and investigated the performance of BP algorithm on four different multi-core platforms, namely, Intel Core 2 Quad, Intel Core I7, PS3 cell and NVIDIA 8800 GTS. In addition, medical imaging applications have witnessed several implementation of CS reconstruction algorithms on multi-core platforms [156–159]. More recently, Bortolotti et al. presented a hardware implementation of OMP on a heterogeneous mobile SoC based on the ARM BIG.LITTLE TM architecture for CS-based ECG monitoring [160]. The proposed architecture shows to be able to provide a real-time reconstruction for ECG signals which can fit well into IoT applications for long-term healthcare monitoring.

5.4. Data Security

Enabling secure communications is an imperative task in IoT applications. In the design of IoT platforms, the sensitive data (image, video, Medical records, etc.) transmitted over the wireless channel have to be secured against unauthorized access. Subsequently, data privacy is a paramount challenge that has to be addressed in IoT applications. A reliable solution can be realized by means of sophisticated state-of-the-art data encryption techniques. Nevertheless, CS can also be well suited to act as an encryption mechanism by exploiting the fact that randomly mapping the original data to another one would encrypt the information. Exploring CS as an encryption technique by using the random sensing matrix as the key has been investigated in [161], where the authors showed that even-though a perfect secrecy is not guaranteed, computational secrecy can be achieved as long as the sensing matrix satisfies the RIP and the number of random projections is at least twice the sparsity of the signal. In addition, Kailkhura et al. [162] showed that the design of an optimal sensing matrix in the context of full secrecy is not universal but it depends mainly on the structure of the signal of interest.

In addition, WSNs have witnessed some applications where CS has been used as both acquisition and encryption scheme. Wang et al. proposed a two-level CS-based security platform to transmit the data from the sensors to the cloud [163]. The first level of encryption is performed by acquiring data following the CS concept. Afterwards, the sensors further encrypt the compressed data without changing its dimension so no additional bandwidth cost is required. Recently, Peng et al. [164] proposed a chaotic CS (CCS) scheme to tackle both energy saving and data security problems in large-scale wireless body area networks (WBANs). The aforementioned schemes are designed mainly to prevent external access to the data. However, to face attacks that can occur within the system (inference attacks) such as controllable event triggering attacks (CETA) and random event triggering attacks (RETA), Hu et al. [165] presented a method called secure compressive data gathering (SCDG). The presented approach is based on changing the sensing matrix coefficients at each time slot.

In addition, cloud security present as well a hot topic to address [166] and CS techniques have shown to disclose great opportunities. For instance, a CS-based encryption scheme for crowd sensing to enable efficient remote sensing system (RSS) maps are presented in [167]. In addition, Wang et al. [168] presented a CS-based framework for preserving the privacy of reconstructed image when operating in outsourcing mode. Cloud outsourcing refers to shifting the data management from the embedded servers to the cloud servers. This framework would be very beneficial for application of connected health-care where the medical images with diagnostic results for different patients are privacy-sensitive. Reader should refer to [169–173] for variety of CS-based frameworks to enable secure cloud operations.

6. Conclusions

IoT applications require a huge number of heterogeneous smart devices to sense, communicate and collaborate with each other in order to gather the maximum information to enable numerous types of services. A large number of sensors will generate big redundant data that would cause unnecessary network traffic, thus degrading the overall system performance. Therefore, exploring CS can reduce the number of data collected and can release the pressure on the wireless communication.

This paper provides a brief overview on CS theory, starting from the basic model of the signal acquisition to the criteria that should be satisfied while designing the sensing matrix, as well as listing some of the most considered reconstruction algorithms used in the literature.

Moreover, this paper reviews the extant work presented in the literature that addresses the integration of CS techniques in the IoT platforms. To this end, and aiming to make the discussion clear and easy to follow, the IoT platform is divided into three layers and the related CS researches for each layer is discussed thoroughly.

The acquisition layer consists of the huge number of sensors that communicate with each other and with a central fusion node. High power consumption presents the most important issue to tackle. Thus, CS techniques can be deployed as a compression technique to reduce the power consumption.

In addition, compressive gathering, sparse networking and hybrid joint DCS-routing protocols can also be explored to achieve energy-efficient sensing layer.

At the processing unit, implementing edge computing platform on FPGAs presents an appealing solution to achieve real-time performance. Several design and architectures for CS reconstruction algorithms have already been proposed and investigated in the literature.

For the application layer, the utilization of CS can be extended from being just a compression paradigm to a broad range of applications by exploring the recovery algorithms associated with CS to enable several applications related to smart cities applications and next generation communication systems.

While it is hard to cover all of the CS software and hardware research that can be integrated with the realm of IoT, this paper highlights the open issues and future trends that can be addressed exploring CS techniques along with some powerful computing platforms to stimulate new research orientation in the field of IoT. The realization of CS encoder that provides directly a compressed signal acquisition is a hot topic to investigate as it enables the next era of digital sampling. In addition, using structured sensing matrices would simplify the acquisition operation and allow real-time data reconstruction. Moreover, providing a CS implementation on commercial multi-core platforms can be well fitted for different IoT applications related to smart home and health care. At the top of all that, CS can be used to enable secure and encrypted transmission scheme by exploring the randomness accommodated with acquisition process without introducing any additional processing.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Alaba, F.A.; Othman, M.; Hashem, I.A.T.; Alotaibi, F. Internet of things security: A survey. *J. Netw. Comput. Appl.* **2017**, *88*, 10–28. [CrossRef]
2. Kortuem, G.; Kawsar, F.; Sundramoorthy, V.; Fitton, D. Smart objects as building blocks for the Internet of things. *IEEE Internet Comput.* **2010**, *14*, 44–51. [CrossRef]
3. Holler, J.; Tsiatsis, V.; Mulligan, C.; Avesand, S.; Karnouskos, S.; Boyle, D. *From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence*; Academic Press: Cambridge, MA, USA, 2014.
4. Lazarescu, M.T. Design of a WSN platform for long-term environmental monitoring for IoT applications. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2013**, *3*, 45–54. [CrossRef]
5. Yang, G.; Xie, L.; Mäntysalo, M.; Zhou, X.; Pang, Z.; Da Xu, L.; Kao-Walter, S.; Chen, Q.; Zheng, L.R. A health-iot platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box. *IEEE Trans. Ind. Inform.* **2014**, *10*, 2180–2191. [CrossRef]
6. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306. [CrossRef]
7. Candès, E.J.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509. [CrossRef]
8. Zhang, Z.; Jung, T.P.; Makeig, S.; Rao, B.D. Compressed sensing for energy-efficient wireless telemonitoring of noninvasive fetal ECG via block sparse Bayesian learning. *IEEE Trans. Biomed. Eng.* **2013**, *60*, 300–309. [CrossRef] [PubMed]
9. Jaspan, O.N.; Fleysher, R.; Lipton, M.L. Compressed sensing MRI: A review of the clinical literature. *Br. J. Radiol.* **2015**, *88*, 20150487. [CrossRef] [PubMed]
10. Djelouat, H.; Baali, H.; Amira, A.; Bensaali, F. IoT Based Compressive Sensing for ECG Monitoring. In Proceedings of the 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Exeter, UK, 21–23 June 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 183–189.

11. Djelouat, H.; Baali, H.; Amira, A.; Bensaali, F. An Adaptive Joint Sparsity Recovery for Compressive Sensing Based EEG System. *Wirel. Commun. Mob. Comput.* **2017**, *2017*, 9823684. [CrossRef]

12. Djelouat, H.; Zhai, X.; Al Disi, M.; Amira, A.; Bensaali, F. System-on-Chip Solution for Patients Biometric: A Compressive Sensing-Based Approach. *IEEE Sens. J.* **2018**. [CrossRef]

13. Duarte-Carvajalino, J.M.; Sapiro, G. Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization. *IEEE Trans. Image Process.* **2009**, *18*, 1395–1408. [CrossRef] [PubMed]

14. Li, S.; Da Xu, L.; Wang, X. Compressed sensing signal and data acquisition in wireless sensor networks and Internet of things. *IEEE Trans. Ind. Inform.* **2013**, *9*, 2177–2186. [CrossRef]

15. Karakus, C.; Gurbuz, A.C.; Tavli, B. Analysis of energy efficiency of compressive sensing in wireless sensor networks. *IEEE Sens. J.* **2013**, *13*, 1999–2008. [CrossRef]

16. Fazel, F.; Fazel, M.; Stojanovic, M. Random access compressed sensing for energy-efficient underwater sensor networks. *IEEE J. Sel. Areas Commun.* **2011**, *29*, 1660–1670. [CrossRef]

17. *ITU-T Y.2060: Overview of the Internet of Things*; ITU: Geneva, Switzerland, 2012.

18. Da Xu, L.; He, W.; Li, S. Internet of things in industries: A survey. *IEEE Trans. Ind. Inform.* **2014**, *10*, 2233–2243.

19. Autili, M.; Inverardi, P.; Tivoli, M. Choreography realizability enforcement through the automatic synthesis of distributed coordination delegates. *Sci. Comput. Program.* **2018**, *160*, 3–29. [CrossRef]

20. Zahariadis, T.; Papadimitriou, D.; Tschofenig, H.; Haller, S.; Daras, P.; Stamoulis, G.D.; Hauswirth, M. Towards a future Internet architecture. In *The Future Internet Assembly*; Springer: Beilin, Germany, 2011; pp. 7–18.

21. Miorandi, D.; Sicari, S.; De Pellegrini, F.; Chlamtac, I. Internet of things: Vision, applications and research challenges. *Ad Hoc Netw.* **2012**, *10*, 1497–1516. [CrossRef]

22. Domingo, M.C. An overview of the Internet of Things for people with disabilities. *J. Netw. Comput. Appl.* **2012**, *35*, 584–596. [CrossRef]

23. Jia, X.; Feng, Q.; Fan, T.; Lei, Q. RFID technology and its applications in Internet of Things (IoT). In Proceedings of the 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), Yichang, China, 21–23 April 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 1282–1285.

24. Liu, C.H.; Yang, B.; Liu, T. Efficient naming, addressing and profile services in Internet-of-Things sensory environments. *Ad Hoc Netw.* **2014**, *18*, 85–101. [CrossRef]

25. Wu, M.; Lu, T.J.; Ling, F.Y.; Sun, J.; Du, H.Y. Research on the architecture of Internet of things. In Proceedings of the 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), Chengdu, China, 20–22 August 2010; IEEE: Piscataway, NJ, USA, 2010; Volume 5, pp. V5–V484.

26. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [CrossRef]

27. Schaffers, H.; Komninos, N.; Pallot, M.; Trousse, B.; Nilsson, M.; Oliveira, A. Smart cities and the future Internet: Towards cooperation frameworks for open innovation. In *The Future Internet Assembly*; Springer: Beilin, Germany, 2011; pp. 431–446.

28. Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. Internet of things for smart cities. *IEEE Internet Things J.* **2014**, *1*, 22–32. [CrossRef]

29. He, W.; Yan, G.; Da Xu, L. Developing vehicular data cloud services in the IoT environment. *IEEE Trans. Ind. Inform.* **2014**, *10*, 1587–1595. [CrossRef]

30. Chunli, L. Intelligent transportation based on the Internet of things. In Proceedings of the 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), Yichang, China, 21–23 April 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 360–362.

31. Huang, L.; Liu, C. The application mode in urban transportation management based on Internet of things. In Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering, ICCSEE, Hangzhou, China, 22–23 March 2013; pp. 1226–1229.

32. Ji, Z.; Anwen, Q. The application of Internet of things(IOT) in emergency management system in China. In Proceedings of the 2010 IEEE International Conference on Technologies for Homeland Security (HST), Waltham, MA, USA, 8–10 November 2010; pp. 139–142.

33. Liu, Z.; Yan, T. Study on multi-view video based on IOT and its application in intelligent security system. In Proceedings of the 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC), Shenyang, China, 20–22 December 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1437–1440.

34. Chianese, A.; Piccialli, F. Designing a smart museum: When cultural heritage joins IoT. In Proceedings of the 2014 Eighth International Conference on Next Generation Mobile Apps, Services and Technologies (NGMAST), Oxford, UK, 10–12 September 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 300–306.

35. Kos, A.; Sedlar, U.; Sterle, J.; Volk, M.; Bešter, J.; Bajec, M. Network monitoring applications based on IoT system. Network and Optical Communications (NOC). In Proceedings of the 18th European Conference on Network and Optical Communications (NOC 2013), 8th Conference on Optical Cabling and Infrastructure (OC&I 2013), Graz, Austria, 10–12 July 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 69–74.

36. Chen, F.; Chandrakasan, A.P.; Stojanovic, V.M. Design and analysis of a hardware-efficient compressed sensing architecture for data compression in wireless sensors. *IEEE J. Solid-State Circuits* **2012**, *47*, 744–756. [CrossRef]

37. Bellasi, D.E.; Benini, L. Energy-efficiency analysis of analog and digital compressive sensing in wireless sensors. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2015**, *62*, 2718–2729. [CrossRef]

38. Donoho, D.L.; Huo, X. Uncertainty principles and ideal atomic decomposition. *IEEE Trans. Inf. Theory* **2001**, *47*, 2845–2862. [CrossRef]

39. Muthukrishnan, S. *Data Streams: Algorithms and Applications*; Now Publishers Inc.: Delft, The Netherlands, 2005.

40. Chen, S.S.; Donoho, D.L.; Saunders, M.A. Atomic decomposition by basis pursuit. *SIAM J. Sci. Comput.* **1998**, *20*, 33–61. [CrossRef]

41. Candes, E.J. The restricted isometry property and its implications for compressed sensing. *Comptes Rendus Math.* **2008**, *346*, 589–592. [CrossRef]

42. Candes, E.; Tao, T. The Dantzig selector: Statistical estimation when p is much larger than n. *Ann. Stat.* **2007**, *35*, 2313–2351. [CrossRef]

43. Figueiredo, M.A.; Nowak, R.D.; Wright, S.J. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. *IEEE J. Sel. Top. Signal Process.* **2007**, *1*, 586–597. [CrossRef]

44. Mallat, S.G.; Zhang, Z. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Process.* **1993**, *41*, 3397–3415. [CrossRef]

45. Krstulovic, S.; Gribonval, R. MPTK: Matching pursuit made tractable. In Proceedings of the 2006 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toulouse, France, 14–19 May 2006; IEEE: Piscataway, NJ, USA, 2006; Volume 3, p. III.

46. Tropp, J.A.; Gilbert, A.C. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Trans. Inf. Theory* **2007**, *53*, 4655–4666. [CrossRef]

47. Needell, D.; Tropp, J.A. CoSaMP: Iterative signal recovery from incomplete and inaccurate samples. *Appl. Comput. Harmon. Anal.* **2009**, *26*, 301–321. [CrossRef]

48. Dai, W.; Milenkovic, O. Subspace pursuit for compressive sensing signal reconstruction. *IEEE Trans. Inf. Theory* **2009**, *55*, 2230–2249. [CrossRef]

49. Needell, D.; Vershynin, R. Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit. *Found. Comput. Math.* **2009**, *9*, 317–334. [CrossRef]

50. Donoho, D.L.; Tsaig, Y.; Drori, I.; Starck, J.L. Sparse solution of underdetermined systems of linear equations by stagewise orthogonal matching pursuit. *IEEE Trans. Inf. Theory* **2012**, *58*, 1094–1121. [CrossRef]

51. Liu, E.; Temlyakov, V.N. The orthogonal super greedy algorithm and applications in compressed sensing. *IEEE Trans. Inf. Theory* **2012**, *58*, 2040–2047. [CrossRef]

52. Blanchard, J.D.; Tanner, J. Performance comparisons of greedy algorithms in compressed sensing. *Numer. Linear Algebra Appl.* **2015**, *22*, 254–282. [CrossRef]

53. Cartis, C.; Thompson, A. A new and improved quantitative recovery analysis for iterative hard thresholding algorithms in compressed sensing. *IEEE Trans. Inf. Theory* **2015**, *61*, 2019–2042. [CrossRef]

54. Foucart, S.; Rauhut, H. *A Mathematical Introduction to Compressive Sensing*; Birkhäuser: Basel, Switzerland, 2013; Volume 1.

55.   Avonds, Y.; Liu, Y.; Van Huffel, S. Simultaneous greedy analysis pursuit for compressive sensing of multi-channel ECG signals. In Proceedings of the 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Chicago, IL, USA, 6–30 August 2014; IEEE: Piscataway, NJ, USA, pp. 6385–6388.

56.   Baron, D.; Duarte, M.F.; Wakin, M.B.; Sarvotham, S.; Baraniuk, R.G. Distributed compressive sensing. *arXiv* **2009**, arXiv:0901.3403.

57.   Tropp, J.A. Algorithms for simultaneous sparse approximation. Part II: Convex relaxation. *Signal Process.* **2006**, *86*, 589–602. [CrossRef]

58.   Tropp, J.A.; Gilbert, A.C.; Strauss, M.J. Algorithms for simultaneous sparse approximation. Part I: Greedy pursuit. *Signal Process.* **2006**, *86*, 572–588. [CrossRef]

59.   He, D.; Zeadally, S. An analysis of RFID authentication schemes for Internet of things in healthcare environment using elliptic curve cryptography. *IEEE Internet Things J.* **2015**, *2*, 72–83. [CrossRef]

60.   Rogers, A.; Jones, E.; Oleynikov, D. Radio frequency identification (RFID) applied to surgical sponges. *Surg. Endosc.* **2007**, *21*, 1235–1237. [CrossRef] [PubMed]

61.   Amendola, S.; Lodato, R.; Manzari, S.; Occhiuzzi, C.; Marrocco, G. RFID technology for IoT-based personal healthcare in smart spaces. *IEEE Internet Things J.* **2014**, *1*, 144–152. [CrossRef]

62.   Khan, M.S.; Islam, M.S.; Deng, H. Design of a reconfigurable RFID sensing tag as a generic sensing platform toward the future Internet of things. *IEEE Internet Things J.* **2014**, *1*, 300–310. [CrossRef]

63.   Attaran, A.; Rashidzadeh, R. Chipless Radio Frequency Identification Tag for IoT Applications. *IEEE Internet Things J.* **2016**, *3*, 1310–1318. [CrossRef]

64.   Islam, S.M.R.; Kwak, D.; Kabir, M.H.; Hossain, M.; Kwak, K.S. The Internet of Things for Health Care: A Comprehensive Survey. *IEEE Access* **2015**, *3*, 678–708. [CrossRef]

65.   Mainetti, L.; Patrono, L.; Vilei, A. Evolution of wireless sensor networks towards the Internet of things: A survey. In Proceedings of the 2011 19th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, 15–17 September 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1–6.

66.   Christin, D.; Reinhardt, A.; Mogre, P.S.; Steinmetz, R. Wireless sensor networks and the Internet of things: Selected challenges. In Proceedings of the 2009 8th GI/ITG Kuvs FachgesprÄch Drahtlose Sensornetze, Hamburg, Germany, 13–14 August 2009; pp. 31–34.

67.   Colitti, W.; Steenhaut, K.; De Caro, N. Integrating wireless sensor networks with the web. In Proceedings of the Extending Internet Low Power Lossy Netw (IP+ SN 2011), Chicago, IL, USA, 11 April 2011.

68.   Harrison, R.R.; Watkins, P.T.; Kier, R.J.; Lovejoy, R.O.; Black, D.J.; Greger, B.; Solzbacher, F. A Low-Power Integrated Circuit for a Wireless 100-Electrode Neural Recording System. *IEEE J. Solid-State Circuits* **2007**, *42*, 123–133. [CrossRef]

69.   Razzaque, M.A.; Bleakley, C.; Dobson, S. Compression in wireless sensor networks: A survey and comparative evaluation. *ACM Trans. Sens. Netw. (TOSN)* **2013**, *10*, 5. [CrossRef]

70.   Verma, N.; Shoeb, A.; Bohorquez, J.; Dawson, J.; Guttag, J.; Chandrakasan, A.P. A micro-power EEG acquisition SoC with integrated feature extraction processor for a chronic seizure detection system. *IEEE J. Solid-State Circuits* **2010**, *45*, 804–816. [CrossRef]

71.   Mamaghanian, H.; Khaled, N.; Atienza, D.; Vandergheynst, P. Compressed sensing for real-time energy-efficient ECG compression on wireless body sensor nodes. *IEEE Trans. Biomed. Eng.* **2011**, *58*, 2456–2466. [CrossRef] [PubMed]

72.   Gedik, B.; Liu, L.; Yu, P.S. ASAP: An Adaptive Sampling Approach to Data Collection in Sensor Networks. *IEEE Trans. Parallel Distrib. Syst.* **2007**, *18*, 1766–1783. [CrossRef]

73.   Abate, F.; Huang, V.K.L.; Monte, G.; Paciello, V.; Pietrosanto, A. A Comparison Between Sensor Signal Preprocessing Techniques. *IEEE Sens. J.* **2015**, *15*, 2479–2487. [CrossRef]

74.   Monte, G. Sensor signal preprocessing techniques for analysis and prediction. In Proceedings of the 2008 34th Annual Conference of IEEE Industrial Electronics, IECON 2008, Orlando, FL, USA, 11–13 November 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 1788–1793.

75.   Monte, G.; Huang, V.; Liscovsky, P.; Marasco, D.; Agnello, A. Standard of things, first step: Understanding and normalizing sensor signals. In Proceedings of the 2013 39th Annual Conference of the IEEE Industrial Electronics Society, IECON, Vienna, Austria, 10–13 November 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 118–123.

76.   Fragkiadakis, A.; Charalampidis, P.; Tragos, E. Adaptive compressive sensing for energy efficient smart objects in IoT applications. In Proceedings of the 2014 4th International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE), Aalborg, Denmark, 11–14 May 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 1–5.

77.   Charalampidis, P.; Fragkiadakis, A.G.; Tragos, E.Z. Rate-adaptive compressive sensing for iot applications. In Proceedings of the 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), Glasgow, Scotland, 11–14 May 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–5.

78.   Peel, L.; Clauset, A. Detecting change points in the large-scale structure of evolving networks. *arXiv* **2014**, arXiv:1403.0989.

79.   Lee, H.; Oh, H.; Lee, S.; Bovik, A.C. Visually weighted compressive sensing: Measurement and reconstruction. *IEEE Trans. Image Process.* **2013**, *22*, 1444–1455. [PubMed]

80.   Lee, H.; Jo, S.K.; Lee, N.; Lee, H.W. A method for co-existing heterogeneous IoT environments based on compressive sensing. In Proceedings of the 2016 18th International Conference on Advanced Communication Technology (ICACT), Pyeongchang, Korea, 31 January–31 February 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 206–209.

81.   Luo, C.; Wu, F.; Sun, J.; Chen, C.W. Efficient Measurement Generation and Pervasive Sparsity for Compressive Data Gathering. *IEEE Trans. Wirel. Commun.* **2010**, *9*, 3728–3738. [CrossRef]

82.   Xu, L.; Qi, X.; Wang, Y.; Moscibroda, T. Efficient data gathering using Compressed Sparse Functions. In Proceedings of the 2013 Proceedings IEEE INFOCOM, Turin, Italy, 14–19 April 2013; pp. 310–314.

83.   Xiang, L.; Luo, J.; Vasilakos, A. Compressed data aggregation for energy efficient wireless sensor networks. In Proceedings of 2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), Salt Lake City, UT, USA, 27–30 June 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 46–54.

84.   Liu, X.Y.; Zhu, Y.; Kong, L.; Liu, C.; Gu, Y.; Vasilakos, A.V.; Wu, M.Y. CDC: Compressive data collection for wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **2015**, *26*, 2188–2197. [CrossRef]

85.   Biswas, S.; Morris, R. Opportunistic routing in multi-hop wireless networks. *ACM Sigcomm Comput. Commun. Rev.* **2004**, *34*, 69–74. [CrossRef]

86.   Du, R.; Gkatzikis, L.; Fischione, C.; Xiao, M. Energy efficient sensor activation for water distribution networks based on compressive sensing. *IEEE J. Sel. Areas Commun.* **2015**, *33*, 2997–3010. [CrossRef]

87.   Chen, W.; Wassell, I.J. Optimized node selection for compressive sleeping wireless sensor networks. *IEEE Trans. Veh. Technol.* **2016**, *65*, 827–836. [CrossRef]

88.   Cao, D.Y.; Yu, K.; Zhuo, S.G.; Hu, Y.H.; Wang, Z. On the implementation of compressive sensing on wireless sensor network. In Proceedings of the 2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI), Berlin, Germany, 4–8 April 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 229–234.

89.   Zhang, B.; Liu, Y.; He, J.; Zou, Z. An energy efficient sampling method through joint linear regression and compressive sensing. In Proceedings of the 2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP), Beijing, China, 9–11 June 2013; pp. 447–450.

90.   Lee, S.; Pattem, S.; Sathiamoorthy, M.; Krishnamachari, B.; Ortega, A. *Compressed Sensing and Routing in Multi-Hop Networks*; University of Southern California CENG Technical Report; University of Southern California: Los Angeles, CA, USA, 2009.

91.   Caione, C.; Brunelli, D.; Benini, L. Distributed compressive sampling for lifetime optimization in dense wireless sensor networks. *IEEE Trans. Ind. Inform.* **2012**, *8*, 30–40. [CrossRef]

92.   Masoum, A.; Meratnia, N.; Havinga, P.J. A distributed compressive sensing technique for data gathering in wireless sensor networks. *Procedia Comput. Sci.* **2013**, *21*, 207–216. [CrossRef]

93.   Zhang, B.; Mor, N.; Kolb, J.; Chan, D.S.; Lutz, K.; Allman, E.; Wawrzynek, J.; Lee, E.A.; Kubiatowicz, J. The Cloud is Not Enough: Saving IoT from the Cloud. In Proceedings of the HotCloud 2015, Santa Clara, CA, USA, 6–7 July 2015.

94.   Datta, S.K.; Bonnet, C.; Nikaein, N. An IoT gateway centric architecture to provide novel M2M services. In Proceedings of the 2014 IEEE World Forum on Internet of Things (WF-IoT), Seoul, Korea, 6–8 March 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 514–519.

95.   Piyare, R. Internet of things: Ubiquitous home control and monitoring system using android based smart phone. *Int. J. Internet Things* **2013**, *2*, 5–11.

96. Samie, F.; Bauer, L.; Henkel, J. IoT technologies for embedded computing: A survey. In Proceedings of the 2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES + ISSS), Pittsburgh, PA, USA, 2–7 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–10.

97. Brzoza-Woch, R.; Nawrocki, P. FPGA-Based Web Services–Infinite Potential or a Road to Nowhere? *IEEE Internet Comput.* **2016**, *20*, 44–51. [CrossRef]

98. Septimus, A.; Steinberg, R. Compressive sampling hardware reconstruction. In Proceedings of the 2010 IEEE International Symposium on Circuits and Systems (ISCAS), Paris, France, 30 May–2 June 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 3316–3319.

99. Stanislaus, J.L.; Mohsenin, T. High performance compressive sensing reconstruction hardware with QRD process. In Proceedings of the 2012 IEEE International Symposium on Circuits and Systems (ISCAS), Seoul, Korea, 20–23 May 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 29–32.

100. Bai, L.; Maechler, P.; Muehlberghuber, M.; Kaeslin, H. High-speed compressed sensing reconstruction on FPGA using OMP and AMP. In Proceedings of the 2012 19th IEEE International Conference on Electronics, Circuits and Systems (ICECS), Seville, Spain, 9–12 December 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 53–56.

101. Ren, F.; Dorrace, R.; Xu, W.; Marković, D. A single-precision compressive sensing signal reconstruction engine on FPGAs. In Proceedings of the 2013 23rd International Conference on Field Programmable Logic and Applications (FPL), Porto, Portugal, 2–4 September 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1–4.

102. Rabah, H.; Amira, A.; Mohanty, B.K.; Almaadeed, S.; Meher, P.K. FPGA implementation of orthogonal matching pursuit for compressive sensing reconstruction. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2015**, *23*, 2209–2220. [CrossRef]

103. Kulkarni, A.; Jafari, A.; Shea, C.; Mohsenin, T. CS-based secured big data processing on FPGA. In Proceedings of the 2016 IEEE 24th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), Washington, DC, USA, 1–3 May 2016; IEEE: Piscataway, NJ, USA, 2016; p. 201.

104. Kulkarni, A.; Stanislaus, J.L.; Mohsenin, T. Parallel heterogeneous architectures for efficient omp compressive sensing reconstruction. In Proceedings of the 2014 SPIE Sensing Technology + Applications, International Society for Optics and Photonics, Baltimore, MD, USA, 5–9 May 2014; p. 91090G.

105. Quan, Y.; Li, Y.; Gao, X.; Xing, M. FPGA Implementation of real-time Compressive Sensing with partial Fourier Dictionary. *Int. J. Antennas Propag.* **2016**, *2016*. [CrossRef]

106. Borghi, A.; Darbon, J.; Peyronnet, S.; Chan, T.F.; Osher, S. A simple compressive sensing algorithm for parallel many-core architectures. *J. Signal Process. Syst.* **2013**, *71*, 1–20. [CrossRef]

107. El-Sayed, M.; Koch, P.; Moullec, Y.L. Architectural design space exploration of an FPGA-based compressed sampling engine: Application to wireless heart-rate monitoring. In Proceedings of the 2015 Nordic Circuits and Systems Conference (NORCAS): NORCHIP International Symposium on System-on-Chip (SoC), Oslo, Norway, 26–28 October 2015; pp. 1–5.

108. Yu, Z.; Su, J.; Yang, F.; Su, Y.; Zeng, X.; Zhou, D.; Shi, W. Fast compressive sensing reconstruction algorithm on FPGA using Orthogonal Matching Pursuit. In Proceedings of the 2016 IEEE International Symposium on Circuits and Systems (ISCAS), Montreal, QC, Canada, 22–25 May 2016; pp. 249–252.

109. Kulkarni, A.; Mohsenin, T. Low Overhead Architectures for OMP Compressive Sensing Reconstruction Algorithm. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2017**, *64*, 1468–1480. [CrossRef]

110. Maechler, P.; Greisen, P.; Sporrer, B.; Steiner, S.; Felber, N.; Burg, A. Implementation of greedy algorithms for LTE sparse channel estimation. In Proceedings of the 2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers (ASILOMAR), Pacific Grove, CA, USA, 7–10 November 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 400–405.

111. Fang, Y.; Chen, L.; Wu, J.; Huang, B. GPU implementation of orthogonal matching pursuit for compressive sensing. In Proceedings of the 2011 IEEE 17th International Conference on Parallel and Distributed Systems (ICPADS), Tainan, Taiwan, 7–9 December 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 1044–1047.

112. Kulkarni, A.; Mohsenin, T. Accelerating compressive sensing reconstruction OMP algorithm with CPU, GPU, FPGA and domain specific many-core. In Proceedings of the 2015 IEEE International Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, 24–27 May 2015; pp. 970–973.

113.   Blache, P.; Rabah, H.; Amira, A. High level prototyping and FPGA implementation of the orthogonal matching pursuit algorithm. In Proceedings of the 2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA), Montreal, QC, Canada, 2–5 July 2012; pp. 1336–1340.

114.   Zhu, H.; Giannakis, G.B. Exploiting sparse user activity in multiuser detection. *IEEE Trans. Commun.* **2011**, *59*, 454–465. [CrossRef]

115.   Schepker, H.F.; Dekorsy, A. Sparse multi-user detection for CDMA transmission using greedy algorithms. In Proceedings of the 2011 8th International Symposium on Wireless Communication Systems (ISWCS), Aachen, Germany, 6–9 November 2011; IEEE: Piscataway, NJ, USA, 2011; pp. 291–295.

116.   Schepker, H.F.; Dekorsy, A. Compressive sensing multi-user detection with block-wise orthogonal least squares. In Proceedings of the 2012 IEEE 75th Vehicular Technology Conference (VTC Spring), Yokohama, Japan, 6–9 May 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 1–5.

117.   Choi, J. Joint device identification and data detection in physical layer for many devices in IoT. In Proceedings of the 2016 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, 19–21 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 85–89.

118.   Choi, J. Two-Stage Multiple Access for Many Devices of Unique Identifications over Frequency-Selective Fading Channels. *IEEE Internet Things J.* **2016**, *4*, 16–171. [CrossRef]

119.   Choi, J. Sparse index multiple access for multi-carrier systems with precoding. *J. Commun. Netw.* **2016**, *18*, 439–445.

120.   Choi, J.W.; Shim, B. Sparse Detection of Non-Sparse Signals for Large-Scale Wireless Systems. *arXiv* **2015**, arXiv:1512.01683.

121.   Wang, B.; Mir, T.; Jiao, R.; Dai, L. Dynamic multi-user detection based on structured compressive sensing for IoT-oriented 5G systems. In Proceedings of the 2016 URSI Asia-Pacific Radio Science Conference (URSI AP-RASC), Seoul, Korea, 21–25 August 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 431–434.

122.   Liu, J.; Cheng, H.Y.; Liao, C.C.; Wu, A.Y.A. Scalable compressive sensing-based multi-user detection scheme for Internet-of-Things applications. In Proceedings of the 2015 IEEE Workshop on Signal Processing Systems (SiPS), Hangzhou, China, 14–16 October 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–6.

123.   Monsees, F.; Bockelmann, C.; Wübben, D.; Dekorsy, A. Sparsity aware multiuser detection for machine to machine communication. In Proceedings of the 2012 IEEE Globecom Workshops (GC Wkshps), Anaheim, CA, USA, 3–7 December 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 1706–1711.

124.   Bockelmann, C.; Schepker, H.F.; Dekorsy, A. Compressive sensing based multi-user detection for machine-to-machine communication. *Trans. Emerg. Telecommun. Technol.* **2013**, *24*, 389–400. [CrossRef]

125.   Swirszcz, G.; Abe, N.; Lozano, A.C. Grouped orthogonal matching pursuit for variable selection and prediction. In Proceedings of the 2009 Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 7–10 December 2009, pp. 1150–1158.

126.   Anagnostopoulos, T.; Zaslavsy, A.; Medvedev, A.; Khoruzhnicov, S. Top—k Query Based Dynamic Scheduling for IoT-enabled Smart City Waste Collection. In Proceedings of the 2015 16th IEEE International Conference on Mobile Data Management, Pittsburgh, PA, USA, 15–18 June 2015; Volume 2, pp. 50–55.

127.   Zhang, J.; Yan, Y.; Chen, L.J.; Wang, M.; Moscibroda, T.; Zhang, Z. Impression Store: Compressive Sensing-based Storage for Big Data Analytics. In Proceedings of the HotCloud 2014, Philadelphia, PA, USA, 17–20 June 2014.

128.   Chen, Y.S.; Tsou, Y.T. Compressive Sensing-Based Adaptive Top-k Query over Compression Domain in Wireless Sensor Networks. In Proceedings of the 2017 IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, CA, USA, 19–22 March 2017; pp. 1–6.

129.   Wang, L.; Zhang, D.; Wang, Y.; Chen, C.; Han, X.; M'hamed, A. Sparse mobile crowdsensing: challenges and opportunities. *IEEE Commun. Mag.* **2016**, *54*, 161–167. [CrossRef]

130.   Hsieh, H.P.; Lin, S.D.; Zheng, Y. Inferring air quality for station location recommendation based on urban big data. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, Australia, 10–13 August 2015; ACM: New York, NY, USA, 2015; pp. 437–446.

131.   Kong, L.; Xia, M.; Liu, X.Y.; Wu, M.Y.; Liu, X. Data loss and reconstruction in sensor networks. In Proceedings of the 2013 Proceedings IEEE INFOCOM, Turin, Italy, 14–19 April 2013; pp. 1654–1662.

132. Wang, L.; Zhang, D.; Yang, D.; Lim, B.Y.; Ma, X. Differential location privacy for sparse mobile crowdsensing. In Proceedings of the 2016 IEEE 16th International Conference on Data Mining (ICDM), Barcelona, Spain, 12–15 December 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1257–1262.

133. Wang, L.; Zhang, D.; Pathak, A.; Chen, C.; Xiong, H.; Yang, D.; Wang, Y. CCS-TA: Quality-guaranteed online task allocation in compressive crowdsensing. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Osaka, Japan, 7–11 September 2015; ACM: New York, NY, USA, 2015; pp. 683–694.

134. Chen, H.; Guo, B.; Yu, Z.; Chen, L.; Ma, X. A generic framework for constraint-driven data selection in mobile crowd photographing. *IEEE Internet Things J.* **2017**, *4*, 284–296. [CrossRef]

135. Xu, L.; Hao, X.; Lane, N.D.; Liu, X.; Moscibroda, T. More with less: Lowering user burden in mobile crowdsourcing through compressive sensing. In Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing, Osaka, Japan, 7–11 September 2015; ACM: New York, NY, USA, 2015; pp. 659–670.

136. Yang, B.; Kaul, M.; Jensen, C.S. Using Incomplete Information for Complete Weight Annotation of Road Networks. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 1267–1279. [CrossRef]

137. Zhu, Y.; Li, Z.; Zhu, H.; Li, M.; Zhang, Q. A compressive sensing approach to urban traffic estimation with probe vehicles. *IEEE Trans. Mob. Comput.* **2013**, *12*, 2289–2302. [CrossRef]

138. Liu, Z.; Li, Z.; Li, M.; Xing, W.; Lu, D. Mining Road Network Correlation for Traffic Estimation via Compressive Sensing. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1880–1893. [CrossRef]

139. Najafabadi, M.M.; Villanustre, F.; Khoshgoftaar, T.M.; Seliya, N.; Wald, R.; Muharemagic, E. Deep learning applications and challenges in big data analytics. *J. Big Data* **2015**, *2*, 1. [CrossRef]

140. Nguyen, M.T.; Teague, K.A. Compressive sensing based random walk routing in wireless sensor networks. *Ad Hoc Netw.* **2017**, *54*, 99–110. [CrossRef]

141. Caione, C.; Brunelli, D.; Benini, L. Compressive sensing optimization for signal ensembles in WSNs. *IEEE Trans. Ind. Inform.* **2014**, *10*, 382–392. [CrossRef]

142. Mamaghanian, H.; Khaled, N.; Atienza, D.; Vandergheynst, P. Design and exploration of low-power analog to information conversion based on compressed sensing. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2012**, *2*, 493–501. [CrossRef]

143. Pareschi, F.; Albertini, P.; Frattini, G.; Mangia, M.; Rovatti, R.; Setti, G. Hardware-algorithms co-design and implementation of an analog-to-information converter for biosignals based on compressed sensing. *IEEE Trans. Biomed. Circuits Syst.* **2016**, *10*, 149–162. [CrossRef] [PubMed]

144. Gangopadhyay, D.; Allstot, E.G.; Dixon, A.M.; Natarajan, K.; Gupta, S.; Allstot, D.J. Compressed sensing analog front-end for bio-sensor applications. *IEEE J. Solid-State Circuits* **2014**, *49*, 426–438. [CrossRef]

145. Kirolos, S.; Laska, J.; Wakin, M.; Duarte, M.; Baron, D.; Ragheb, T.; Massoud, Y.; Baraniuk, R. Analog-to-information conversion via random demodulation. In Proceedings of the 2006 IEEE Dallas/CAS Workshop on Design, Applications, Integration and Software, Madrid, Spain, 29–30 October 2006; IEEE: Piscataway, NJ, USA, 2006; pp. 71–74.

146. Chen, X.; Yu, Z.; Hoyos, S.; Sadler, B.M.; Silva-Martinez, J. A sub-Nyquist rate sampling receiver exploiting compressive sensing. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2011**, *58*, 507–520. [CrossRef]

147. Baheti, P.K.; Garudadri, H. An ultra low power pulse oximeter sensor based on compressed sensing. In Proceedings of the 2009 Sixth International Workshop on Wearable and Implantable Body Sensor Networks, BSN 2009, Berkeley, CA, USA, 3–5 June 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 144–148.

148. Candès, E.J.; Wakin, M.B. An introduction to compressive sampling. *IEEE Signal Process. Mag.* **2008**, *25*, 21–30. [CrossRef]

149. Mendelson, S.; Pajor, A.; Tomczak-Jaegermann, N. Uniform uncertainty principle for Bernoulli and subgaussian ensembles. *Constr. Approx.* **2008**, *28*, 277–289. [CrossRef]

150. Candes, E.J.; Tao, T. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Trans. Inf. Theory* **2006**, *52*, 5406–5425. [CrossRef]

151. Davenport, M.A. Random Observations on Random Observations: Sparse Signal Acquisition and Processing. Ph.D. Thesis, Rice University, Houston, TX, USA, 2010.

152. Ravelomanantsoa, A.; Rabah, H.; Rouane, A. Simple deterministic measurement matrix: Application to EMG signals. In Proceedings of the 2014 26th International Conference on Microelectronics (ICM), Doha, Qatar, 14–17 December 2014; pp. 76–79.

153. Bah, B.; Tanner, J. Vanishingly sparse matrices and expander graphs, with application to compressed sensing. *IEEE Trans. Inf. Theory* **2013**, *59*, 7491–7508. [CrossRef]

154. He, Z.; Ogawa, T.; Haseyama, M. The simplest measurement matrix for compressed sensing of natural images. In Proceedings of the 2010 IEEE International Conference on Image Processing, Hong Kong, China, 26–29 September 2010; pp. 4301–4304.

155. Munir, A.; Gordon-Ross, A.; Ranka, S. Multi-core embedded wireless sensor networks: Architecture and applications. *IEEE Trans. Parallel Distrib. Syst.* **2014**, *25*, 1553–1562. [CrossRef]

156. Murphy, M.; Alley, M.; Demmel, J.; Keutzer, K.; Vasanawala, S.; Lustig, M. Fast l1 SPIRiT Compressed Sensing Parallel Imaging MRI: Scalable Parallel Implementation and Clinically Feasible Runtime. *IEEE Trans. Med Imaging* **2012**, *31*, 1250–1262. [CrossRef] [PubMed]

157. Chang, C.H.; Ji, J. Compressed sensing MRI with multichannel data using multicore processors. *Magn. Reson. Med.* **2010**, *64*, 1135–1139. [CrossRef] [PubMed]

158. Smith, D.S.; Gore, J.C.; Yankeelov, T.E.; Welch, E.B. Real-time compressive sensing MRI reconstruction using GPU computing and split Bregman methods. *Int. J. Biomed. Imaging* **2012**, *2012*. [CrossRef] [PubMed]

159. Kim, D.; Trzasko, J.; Smelyanskiy, M.; Haider, C.; Dubey, P.; Manduca, A. High-performance 3D compressive sensing MRI reconstruction using many-core architectures. *J. Biomed. Imaging* **2011**, *2011*, 2. [CrossRef] [PubMed]

160. Bortolotti, D.; Mangia, M.; Bartolini, A.; Rovatti, R.; Setti, G.; Benini, L. Energy-Aware Bio-signal Compressed Sensing Reconstruction on the WBSN-gateway. *IEEE Trans. Emerg. Top. Comput.* **2016**. [CrossRef]

161. Rachlin, Y.; Baron, D. The secrecy of compressed sensing measurements. In Proceedings of the 2008 46th Annual Allerton Conference on Communication, Control, and Computing, Monticello, IL, USA, 23–26 September 2008; pp. 813–817.

162. Kailkhura, B.; Liu, S.; Wimalajeewa, T.; Varshney, P.K. Measurement Matrix Design for Compressed Detection With Secrecy Guarantees. *IEEE Wirel. Commun. Lett.* **2016**, *5*, 420–423. [CrossRef]

163. Wang, C.; Zhang, B.; Ren, K.; Roveda, J.M.; Chen, C.W.; Xu, Z. A privacy-aware cloud-assisted healthcare monitoring system via compressive sensing. In Proceedings of the 2014 IEEE INFOCOM, Toronto, ON, Canada, 27 April–2 May 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 2130–2138.

164. Peng, H.; Tian, Y.; Kurths, J.; Li, L.; Yang, Y.; Wang, D. Secure and Energy-Efficient Data Transmission System Based on Chaotic Compressive Sensing in Body-to-Body Networks. *IEEE Trans. Biomed. Circuits Syst.* **2017**, *11*, 558–573. [CrossRef] [PubMed]

165. Hu, P.; Xing, K.; Cheng, X.; Wei, H.; Zhu, H. Information leaks out: Attacks and countermeasures on compressive data gathering in wireless sensor networks. In Proceedings of the 2014 IEEE INFOCOM, Toronto, ON, Canada, 27 April–2 May 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 1258–1266.

166. Singh, S.; Jeong, Y.S.; Park, J.H. A survey on cloud computing security: Issues, threats, and solutions. *J. Netw. Comput. Appl.* **2016**, *75*, 200–222. [CrossRef]

167. Wu, X.; Yang, P.; Tang, S.; Zheng, X.; Xiong, Y. Privacy preserving RSS map generation for a crowdsensing network. *IEEE Wirel. Commun.* **2015**, *22*, 42–48. [CrossRef]

168. Wang, C.; Zhang, B.; Ren, K.; Roveda, J.M. Privacy-Assured Outsourcing of Image Reconstruction Service in Cloud. *IEEE Trans. Emerg. Top. Comput.* **2013**, *1*, 166–177. [CrossRef]

169. Kang, L.W.; Muchtar, K.; Wei, J.D.; Lin, C.Y.; Chen, D.Y.; Yeh, C.H. Privacy-preserving multimedia cloud computing via compressive sensing and sparse representation. In Proceedings of the 2012 International Conference on Information Security and Intelligence Control (ISIC), Yunlin, Taiwan, 14–16 August 2012; IEEE: Piscataway, NJ, USA, 2012; pp. 246–249.

170. Wang, Q.; Zeng, W.; Tian, J. A compressive sensing based secure watermark detection and privacy preserving storage framework. *IEEE Trans. Image Process.* **2014**, *23*, 1317–1328. [CrossRef] [PubMed]

171. Zhang, Y.; Zhou, J.; Zhang, L.Y.; Chen, F.; Lei, X. Support-set-assured parallel outsourcing of sparse reconstruction service for compressive sensing in multi-clouds. In Proceedings of the 2015 International Symposium on Security and Privacy in Social Networks and Big Data (SocialSec), Hangzhou, China, 16–18 November 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 1–6.

172.   Jyothish, L.G.; Veena, V.; Soman, K.   A cryptographic approach to video watermarking based on compressive sensing, arnold transform, sum of absolute deviation and SVD. In Proceedings of the 2013 Annual International Conference on Emerging Research Areas and 2013 International Conference on Microelectronics, Communications and Renewable Energy (AICERA/ICMiCR), Kerala, India, 4–6 June 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1–5.

173.   Bianchi, T.; Bioglio, V.; Magli, E.   Analysis of one-time random projections for privacy preserving compressed sensing. *IEEE Trans. Inf. Forensics Secur.* **2016**, *11*, 313–327. [CrossRef]